

# **Atualização Tecnológica do Controlo de uma Mesa Hidráulica Móvel**

*Pedro Manuel Nogueira dos Santos*

**Dissertação de Mestrado**

Orientador: Prof. Francisco Jorge Teixeira de Freitas



**Mestrado Integrado em Engenharia Mecânica**

**Opção de Automação**

Setembro de 2018



*The first 90 percent of the code accounts for the first 90 percent of the development time. The remaining 10 percent of the code accounts for the other 90 percent of the development time.*

*— Tom Cargill*



## Resumo

Este trabalho consiste na quarta fase de desenvolvimento de um protótipo de um equipamento robótico móvel que foi desenvolvido para movimentação em ambientes interiores e o seu fim é puramente demonstrativo. Este veículo possui um modo de locomoção inovador, envolvendo patas e rodas, que lhe permite uma grande variedade de movimentos e uma capacidade de carga assinalável.

Nas fases anteriores de desenvolvimento procedeu-se à construção do veículo com acionamento hidráulico, capaz de transportar carga, deslocar-se eficazmente sobre pavimentos e ultrapassar algumas barreiras arquitetónicas, como pequenos degraus, e é dotado de um sistema de direção controlável e de auto nivelção e à implementação de um modo de navegação autónomo baseado em sensores óticos.

Dado que esta fase se encontra oito anos após a última alteração efetuada no protótipo, o principal objetivo foi a atualização tecnológica da Mesa Hidráulica Móvel. Foram substituídos todos os componentes inoperacionais e desatualizados como a fonte primária de energia, o controlador central e reformulada a solução de comunicação da informação. Simultaneamente foi implementado um novo modo de navegação com recurso a visão artificial e complementado por sensores ultrassónicos de proximidade.

Com a instalação de um novo controlador central foi necessário desenvolver de raiz uma aplicação informática de controlo do veículo. Este possui a capacidade de comandar o movimento de todos os atuadores individualmente e em conjunto, manter a horizontalidade da superfície superior do protótipo e conduzir o veículo de forma autónoma ou manual.

Foi também criado um programa para processamento da imagem e leitura dos sensores que é corrido em microcomputadores independentes que comunicam posteriormente a informação necessária ao controlador central.

Finalmente, foi concebida uma interface gráfica, aplicada num computador, que permite ao operador a supervisão e comando da Mesa Hidráulica Móvel de modo remoto através de comunicação *Wi-Fi*.



## Technological Update of the Control of a Mobile Hydraulic Bench

### Abstract

This project consists of the fourth stage of development of a prototype of a mobile robotic equipment which was developed for indoor use and it is purely for demonstration purposes. This vehicle has an innovative locomotion mode, involving legs and wheels, which allows a great variety of movements and load carrying capabilities.

On the previous stages of development, a hydraulically driven vehicle was built which is able to transport some external load, moving efficiently on pavements and overcoming architectural barriers, such as small steps, with a controllable steering system and self-leveling capabilities and a navigation mode based on optical sensors.

Since this phase is taking place eight years after the last works made on the prototype, the main objective was to have a technological update of the Hydraulic Mobile Bench. All inoperative and out-of-date components such as batteries, programmable logic controller and frequency inverter have been replaced. Simultaneously, a new navigation mode was implemented using artificial vision and complemented by ultrasonic proximity sensors.

With the installation of a new controller it was necessary to develop a control program for the vehicle from scratch. It has the ability to control the movement of all actuators individually and jointly, maintain the horizontality of the top surface of the prototype and drive the vehicle autonomously or manually.

A program was also created for image processing and sensor reading that is run on independent microcomputers that later communicate the necessary information to the controller.

Finally, a graphical interface was designed, and applied to a computer, which allows the operator to supervise and control the Mobile Hydraulic Bench remotely via Wi-Fi communication.





## Agradecimentos

Primeiramente, gostaria de agradecer ao meu orientador Professor Doutor Francisco Jorge Teixeira de Freitas pela valiosa partilha de conhecimentos, acompanhamento, compreensão e pela oportunidade de realizar este projeto.

Agradeço ao Sr. Joaquim Silva e ao Sr. Ramalho, técnicos de laboratório, pela disponibilidade e preciosa ajuda no desenvolvimento deste trabalho.

Ao Engenheiro Tiago Abreu, pela camaradagem e auxílio prestado nas questões de programação.

Aos meus colegas e amigos pela cooperação e companheirismo demonstrado.

Quero agradecer à Margarida Pinho pela motivação dada e apoio incansável.

Finalmente, gostaria de agradecer à minha família, especialmente aos meus pais e irmão, sem os quais me teria sido impossível completar esta etapa da minha vida.



## Índice de Conteúdos

1	Introdução .....	1
1.1	Revisão prévia dos trabalhos anteriores .....	1
1.2	Considerações sobre os trabalhos anteriores .....	4
1.3	Objetivos do projeto .....	5
1.4	Organização e temas abordados .....	5
2	Análise preliminar e considerações .....	7
2.1	Estrutura mecânica e acionamentos .....	7
2.2	Sistema hidráulico e fonte de energia .....	9
2.3	Sensorização .....	12
2.4	Sistema elétrico e <i>hardware</i> .....	14
3	Análise de sistemas de navegação.....	19
3.1	Sistemas de navegação .....	19
3.2	Opção escolhida .....	22
4	Conceção e desenvolvimento da nova solução.....	23
4.1	Visão Artificial.....	23
4.2	Sensores de proximidade.....	25
4.3	Suportes para sensores e câmaras .....	28
4.4	Sistema elétrico e <i>hardware</i> .....	30
5	Aplicação informática desenvolvida .....	35
5.1	Comunicação .....	35
5.2	Processamento de imagem e sensores ultrassónicos .....	39
5.3	Programa de controlo dos movimentos do veículo .....	45
5.4	Interface .....	53
6	Conclusão .....	67
6.1	Trabalhos futuros .....	67
	Bibliografia.....	68
ANEXO A:	Desenho de construção.....	71
ANEXO B:	Tabela de ligações ao autómato .....	73
ANEXO C:	Esquemas elétricos .....	77



## Índice de Figuras

Figura 1.1 - Veículo Hidráulico (1ª versão do objeto de estudo) .....	2
Figura 1.2 - Mesa Hidráulica Móvel (2ª versão do objeto de estudo) .....	3
Figura 1.3 - Suporte para sensores óticos .....	4
Figura 2.1 - Articulação de direção .....	7
Figura 2.2 - Sistema de acionamento da direção .....	8
Figura 2.3 - Modelo 3D da perna da Mesa Hidráulica Móvel.....	8
Figura 2.4 - Estrutura mecânica da Mesa Hidráulica Móvel .....	9
Figura 2.5 – Exemplo de central hidráulica compacta .....	9
Figura 2.6 - Sub-circuito hidráulico de acionamento de uma perna .....	10
Figura 2.7 - Sub-circuito de contrapressão .....	11
Figura 2.8 - Sub-circuito de direção .....	11
Figura 2.9 - Sistema hidráulico da Mesa Hidráulica Móvel .....	12
Figura 2.10 - Transdutor de posição <i>Micro-Epsilon MK30</i> .....	12
Figura 2.11 - Transdutor de posição <i>LTH100S</i> .....	13
Figura 2.12 - Transdutor de pressão <i>TP3BE53R</i> .....	13
Figura 2.13 - Inclínometro <i>SCA121T</i> .....	13
Figura 2.14 - Sensor ótico <i>XUB0BPSNM12</i> .....	14
Figura 2.15 - Autômato e módulos de expansão .....	14
Figura 2.16 - Variador de frequência <i>Varispeed-606PC30</i> .....	15
Figura 2.17 - Inversor de corrente <i>PSI100S</i> .....	16
Figura 2.18 - Esquema de alimentação elétrica.....	17
Figura 3.1 - AGV com navegação por filoguiamento .....	20
Figura 3.2 - AGV com navegação por sensores óticos .....	20
Figura 3.3 - Esquema de AGV com navegação por grelha de <i>transponders</i> .....	21
Figura 3.4 - Esquema de AGV com navegação por triangulação.....	21
Figura 4.1 - <i>Raspberry Pi 3</i> modelo B.....	24
Figura 4.2 - <i>Raspberry Pi Camera Module</i> .....	24
Figura 4.3 - Grelha de detecção de proximidade.....	25
Figura 4.4 - Sensor de proximidade ultrassônico HC-SR04 .....	26
Figura 4.5 - Diagrama temporal de funcionamento do HC-SR04 .....	26
Figura 4.6 - Circuito divisor de tensão .....	27
Figura 4.7 - Circuito de ligação do sensor ao microcomputador .....	27
Figura 4.8 - Lista GPIO .....	28
Figura 4.9 - Modelo <i>SolidWorks</i> do suporte .....	29
Figura 4.10 – Fixação inicial do suporte .....	29
Figura 4.11 - Módulo base <i>TM251MESC</i> .....	30
Figura 4.12 - Módulos de expansão.....	31
Figura 4.13 - Esquema elétrico de alimentação atual.....	32
Figura 4.14 - Variador de frequência <i>3G3JV-AB004</i> .....	33

Figura 4.15 - <i>Access Point EAP115Wall</i> .....	34
Figura 4.16 - Injetor PoE TL-PoE150S .....	34
Figura 5.1 - Arquitetura de comunicação .....	36
Figura 5.2 - Configuração do IP fixo no PLC .....	36
Figura 5.3 - Configuração do IP fixo na <i>Raspberry Pi</i> .....	37
Figura 5.4 - Configuração do IP fixo no computador .....	37
Figura 5.5 - <i>Login Omada Controller</i> .....	38
Figura 5.6 - Definição do nome e <i>password</i> da rede <i>Wi-Fi</i> .....	38
Figura 5.7 - Criação de conta no <i>Omada Controller</i> .....	38
Figura 5.8 - Ambiente de trabalho do <i>Raspbian</i> versão <i>Stretch 4</i> .....	39
Figura 5.9 - Teste de detecção do colete refletor .....	41
Figura 5.10 - Ficheiro lido na inicialização .....	45
Figura 5.11 - Identificação dos módulos usados no programa .....	46
Figura 5.12 - Lista ordenada dos subprogramas .....	46
Figura 5.13 – Correspondência das variáveis com os endereços de comunicação.....	47
Figura 5.14 - Relação entre distâncias e alturas reais e aparentes.....	47
Figura 5.15 - <i>Grafcet</i> do modo de funcionamento .....	48
Figura 5.16 - <i>Grafcet</i> de controlo dos motores hidráulicos .....	49
Figura 5.17 - <i>Grafcet</i> de controlo da elevação .....	50
Figura 5.18 - <i>Grafcet</i> de controlo da inclinação .....	50
Figura 5.19 - <i>Grafcet</i> de controlo da pressão .....	51
Figura 5.20 - <i>Grafcet</i> de controlo da direção .....	51
Figura 5.21 - Grelha de comandos baseada nos sinais dos sensores de proximidade .....	52
Figura 5.22 - Página base da interface .....	54
Figura 5.23 - Interface no estado de emergência .....	55
Figura 5.24 - Interface no modo manual .....	56
Figura 5.25 - Inversão do sentido de navegação.....	57
Figura 5.26 - Interface no modo manual avançado .....	58
Figura 5.27 - Interface no modo autónomo.....	59
Figura 5.28 - Diferença entre grelha sem e com deteções, respetivamente .....	60
Figura 5.29 - Interface no estado de emergência por falha na comunicação.....	60
Figura 5.30 - Configuração de <i>Power switch</i> .....	61
Figura 5.31 - Configuração de <i>Meter 180º</i> .....	61
Figura 5.32 - Configuração do <i>Progress bar</i> .....	62
Figura 5.33 - Configuração do <i>SpinControl</i> .....	62
Figura 5.34 - Configuração do <i>Slider</i> .....	63
Figura 5.35 - Configuração do <i>Rectangle</i> para leitura de valores .....	63
Figura 5.36 - Configuração de <i>Image</i> .....	64
Figura 5.37 - Configuração de <i>OnMouseDown</i> .....	64
Figura 5.38 - Configuração de <i>Toggle a Variable</i> .....	65

Figura 5.39 - Configuração de <i>Switch Framevisualization</i> .....	65
Figura 5.40 - Configuração de <i>Frame</i> .....	66
Figura 5.41 - Configuração invisibilidade de <i>Image</i> .....	66





## Índice de Tabelas

Tabela 1 - Funções dos diversos módulos do autómato .....	15
Tabela 2 - Ligações entre <i>Raspberry PI</i> e HC-SR04 .....	28
Tabela 3 - Funções dos diversos módulos do autómato .....	31
Tabela 4 - Parâmetros configurados no variador de frequência.....	33
Tabela 5 - Legenda das funções da página base da interface .....	54
Tabela 6 - Legenda das funções da interface no modo manual.....	56
Tabela 7 - Legenda das funções da interface no modo manual avançado .....	58
Tabela 8 - Legenda das funções da interface no modo autónomo.....	59



# 1 Introdução

A robótica tem evoluído imenso desde o seu aparecimento sendo agora usada não só para tarefas que não podem ser realizadas por humanos, mas também para aumentar o seu conforto e facilitar a sua vida.

No decorrer desta evolução surgiram os Automated Guided Vehicles (AGVs), veículos com a capacidade de se mover sem o comando direto de um operador. É nesta vertente da robótica que o objeto de estudo desta monografia, designado por Mesa Hidráulica Móvel, se integra.

A solução analisada neste projeto possui a particularidade de utilizar em simultâneo patas e rodas (dois meios de locomoção comuns em AGVs), permitindo combinar uma movimentação simples e eficaz em terrenos planos com a capacidade de ultrapassar pequenos obstáculos.

Esta dissertação tem como fim expor a quarta fase do desenvolvimento desta Mesa Hidráulica Móvel, concebida no âmbito da Unidade Curricular de Dissertação, do Mestrado Integrado em Engenharia Mecânica, opção de Automação, na Faculdade de Engenharia da Universidade do Porto, concretizado no segundo semestre do ano letivo de 2017/2018, contando com a orientação do Professor Francisco Freitas.

O principal propósito deste trabalho é analisar o produto das várias fases de desenvolvimento a que este AGV foi sujeito e desenvolver algumas soluções para melhorar o seu estado de desempenho e as suas funcionalidades.

## 1.1 Revisão prévia dos trabalhos anteriores

Este trabalho é, como referido anteriormente, a quarta etapa de desenvolvimento deste veículo hidráulico. Vai ser descrito nesta secção o progresso do mesmo ao longo das três fases que precederam este projeto.

### 1.1.1 Primeira fase - A versão inicial

A Mesa Hidráulica Móvel foi construída de raiz pelo aluno André Martins na unidade curricular de Laboratórios de Automação, disciplina integrante da opção de Automação da Licenciatura em Engenharia Mecânica da Faculdade de Engenharia da Universidade do Porto no ano letivo de 2006/2007, sob a orientação do Professor Francisco Freitas (que viria a orientar todas as restantes fases, incluindo a presente).

Partiu de uma ideia inicial de criar um protótipo de um veículo inovador que circulasse em ambientes interiores, mas que fosse capaz de alguma flexibilidade para fins demonstrativos. Optou-se por uma junção de rodas, que permitem um deslocamento eficaz em superfícies planas, com pernas criando a capacidade de ultrapassar alguns obstáculos.

Nesta primeira fase tinha-se ainda como objetivo que o protótipo não fosse apenas capaz de flexibilidade, mas também de transporte de carga. Tendo em vista esse fim foram escolhidos

acionamentos hidráulicos em oposição a elétricos (normalmente usados neste género de veículos), pois estes permitem maior capacidade de força/binários, particularmente a baixas velocidades, sendo ao mesmo tempo compactos. A alimentação energética do veículo é, contudo, elétrica, possuindo oito baterias de 12 VDC que alimentam todo o protótipo incluindo o motor elétrico que aciona a bomba hidráulica.

Tendo em conta todas as características anteriormente referidas foi construído o protótipo possuindo pernas com dupla articulação e rodas nas extremidades, sendo cada uma acionada pelo seu cilindro ou motor hidráulico, tal como é visível na Figura 1.1.



Figura 1.1 - Veículo Hidráulico (1ª versão do objeto de estudo)<sup>[1]</sup>

O veículo foi também dotado de diversos transdutores, para ler a posição de cada atuador linear, e um inclinómetro de dois graus de liberdade para ser possível medir a variação da altura e a inclinação do mesmo. Deste modo foi possível garantir a horizontalidade da mesa durante o transporte de carga em pavimentos inclinados.

Os atuadores podem ter movimento individual ou em conjunto sendo o seu comando efetuado por um micro autómato programável industrial da *Schneider Electric*, modelo *Twido*.<sup>[1][2]</sup>

### 1.1.2 Segunda fase - A mesa articulada

Posteriormente o protótipo foi o objeto de estudo duma dissertação de fim de curso, realizada no 2º semestre do ano letivo de 2008/2009 pelo aluno Tiago Teixeira, a qual teve por objetivo a melhoria da, agora intitulada, “Mesa Hidráulica Móvel”.

Após a análise da primeira versão da mesa, foram identificadas algumas características que eram dignas de melhorias, mas a ideia base e objetivos iniciais foram mantidos os mesmos.

A primeira alteração surgiu devido à falta de capacidade de mudança de direção, o que limitava os seus deslocamentos a percorrer uma linha reta. Para ultrapassar tal limitação foi adicionada uma segunda mesa e colocado um sistema de direção, tipo *dumper*, por articulação central.

Outro ponto que merecia atenção era o peso próprio da mesa que, devido à utilização de uma central hidráulica desatualizada e a válvulas convencionais instaladas em blocos de ferro fundido, tinha um valor excessivo. Este problema foi solucionado através da instalação de uma central hidráulica mais compacta e à construção de novos blocos em alumínio para a aplicação de válvulas de rosca novas, tanto para substituir os blocos anteriores, bem como para o novo bloco da direção.

O aumento de área da Mesa Móvel e a compactação do sistema hidráulico permitiu que todos os componentes (hidráulicos e elétricos) fossem instalados debaixo das mesas, libertando assim na totalidade o espaço superior das mesas para melhor acomodar carga a transportar, como é possível observar na Figura 1.2. Desta forma foi resolvido o problema do espaço para carga que a solução anterior tinha revelado.

Foram ainda instalados sensores de pressão no atuador de elevação de cada perna de modo a permitir a monitorização da carga na mesma e, ainda mais importante, a identificação de situações em que as pernas possam estar em suspenso no ar, isto é, sem sustentarem o peso próprio do veículo.

Finalmente foram acrescentados módulos ao autómato e este foi reprogramado para permitir a mudança de direção da mesa e a capacidade de manter a horizontalidade de modo automático. Procedeu-se também à criação de uma interface de uso simples e agradável para o utilizador com a possibilidade de comando em modo remoto através da implementação de um sistema de controlo e aquisição de dados e de comunicação *bluetooth*.<sup>[1][2]</sup>



Figura 1.2 - Mesa Hidráulica Móvel (2ª versão do objeto de estudo)<sup>[1]</sup>

### 1.1.3 Terceira fase - A condução autónoma

A última fase de desenvolvimento antes da atual decorreu no ano letivo de 2009/2010 e enquadra-se na dissertação “Controlo de Movimento em Mesa Hidráulica Móvel” realizada por Manuel Ribeiro.

Esta etapa teve como principal objetivo dotar o veículo de um modo de condução autónoma, mantendo os requisitos e as bases das fases anteriores.

Após o estudo dos diversos modos de navegação utilizados nos AGVs (*Automated Guided Vehicles*), foi decidido utilizar uma navegação baseada em sensores óticos e no seguimento de uma fita colorida colocada sobre o pavimento. Esta solução permite definir um

percurso fixo para o veículo que pode ser facilmente alterado, substituído em caso de danificado e não apresenta grandes custos de componentes nem de instalação.

A instalação dos sensores requereu a execução de suportes para os mesmos, garantindo permanentemente uma direção vertical. A solução encontrada está representada na Figura 1.3.



Figura 1.3 - Suporte para sensores óticos<sup>[2]</sup>

Esta evolução levou à necessidade de alterar o sistema elétrico, de forma a alimentar os sensores óticos e a reduzir o tamanho do anterior para manter todos os componentes na parte inferior da mesa.

Foi ainda desenvolvido um novo *software* de controlo acrescentando a capacidade de calibrar a mesa, controlar os atuadores lineares através de referências de posição e comandar o AGV de modo remoto ou autónomo.

Por fim foi desenvolvida uma nova HMI (*Human Machine Interface*) em *Vijeo Citect* para acomodar as alterações feitas ao protótipo.<sup>[2]</sup>

## 1.2 Considerações sobre os trabalhos anteriores

Após as três fases de desenvolvimento realizadas no protótipo, é possível dizer sem dúvidas que este é um AGV funcional, interessante e com potencial para continuado desenvolvimento.

Com uma estrutura de pernas de dupla articulação e rodas nas extremidades a mesa apresenta uma base flexível e eficaz. Estando associado a acionamentos hidráulicos tem também uma capacidade de carga admirável, sendo, portanto, uma excelente base de trabalho.

A nível quantitativo o veículo é capaz de se movimentar nos dois sentidos com velocidades máximas de aproximadamente 5 km/h, podendo também efetuar mudanças de direção com um raio mínimo de curvatura de cerca de 5,8m. Embora a velocidade não se apresente como um problema para o género de trabalhos que se pretende que realize, o seu ângulo de viragem ( $\pm 17^\circ$ ) é relativamente baixo, não permite grande mobilidade. Apresenta ainda uma aptidão de subida e descida da parte superior da mesa entre 600 e 1060 mm, em relação ao nível do chão, e uma capacidade de carga líquida de 100 kg.

A Mesa Móvel tem um sistema de navegação autónoma baseado numa fita preta aplicada no pavimento e sensores óticos que, embora não seja a opção com mais potencialidades, é funcional e adequada para os objetivos previamente estabelecidos.

Uma vez que a última fase de trabalho sobre esta mesa foi há sensivelmente oito anos, é notória a marca do tempo. Devido a tal, o objeto de estudo apresenta um autómato programável

bastante desatualizado, um baixo nível de óleo, baterias inutilizáveis e a ausência do antigo percurso marcado a fita preta.

Após análise mais detalhada foi possível confirmar que, à exceção das baterias, todos os componentes se encontravam operacionais.

### 1.3 Objetivos do projeto

Estando este projeto enquadrado num trabalho de desenvolvimento contínuo, teve-se como intenção manter a maioria dos objetivos globais intactos.

Pretende-se, portanto, que o veículo mantenha a capacidade de garantir a horizontalidade de modo autónomo, tal como a capacidade de superar pequenos obstáculos e de regular o nível da mesa.

Nesta etapa do desenvolvimento do protótipo tem-se como principal objetivo atualizar os componentes obsoletos e dotá-lo de recursos para uma futura evolução contínua.

Pretende-se, simultaneamente, alterar o modo de navegação para uma alternativa com mais potencial e flexibilidade, mantendo-o adequado para o local e tendo em conta os custos envolvidos.

Tem-se ainda como intuito elaborar uma nova aplicação informática de controlo para acomodar as alterações de funcionalidades e as alterações de *hardware*.

Por fim, tem-se como intenção desenvolver uma interface simples e intuitiva, mas eficaz, para o utilizador. Esta manterá a capacidade de comunicação remota com um computador, libertando a mesa de qualquer cabo, mas passará de uma comunicação por *bluetooth* para uma comunicação por *Wi-Fi*.

### 1.4 Organização e temas abordados

O presente documento encontra-se dividido em sete capítulos, contando com este, tendo ainda 3 anexos.

O capítulo seguinte será *Análise preliminar e considerações* e consiste no estudo do estado do protótipo de forma a poder determinar o que melhorar.

O terceiro capítulo denomina-se de *Análise de sistemas de navegação* e neste ir-se-á efetuar uma descrição das várias soluções de navegação usadas atualmente e decidir-se-á qual implementar na mesa.

Designado de *Conceção e desenvolvimento da nova solução*, o quarto capítulo apresenta todas as alterações de *hardware* efetuadas na Mesa Hidráulica Móvel.

O quinto capítulo, denominado de *Software desenvolvido*, descreve-se todos os programas criados para o funcionamento do veículo.

De seguida, são expostas as conclusões retiradas deste trabalho e as propostas para trabalhos futuros no capítulo seis.

Por fim, apresenta-se as referências usadas no desenvolvimento deste projeto.

Os anexos presentes no fim deste trabalho são constituídos pelo desenho de construção do suporte para os sensores e para a câmara, pelas ligações ao autómato e pelo esquema elétrico.





## 2 Análise preliminar e considerações

Neste capítulo pretende-se analisar detalhadamente o estado do objeto de estudo aquando do início da realização desta dissertação, de modo a ser possível entender melhor as suas potencialidades e limitações.

Com esta primeira abordagem é possível definir de modo mais eficiente quais os pontos menos favoráveis e que podem ser melhorados no decorrer deste projeto.

De forma a simplificar a análise da Mesa Móvel este capítulo está dividido nas seguintes secções:

- *Estrutura mecânica e acionamentos*, descrição da estrutura mecânica e dos meios de acionamento;
- *Sistema hidráulico e fonte de energia*, estudo do sistema hidráulico implementado no protótipo;
- *Sensorização*, análise dos diversos sensores e transdutores aplicados no veículo;
- *Sistema elétrico e hardware de comando*, estudo e descrição dos componentes elétricos usados e do sistema elétrico.

### 2.1 Estrutura mecânica e acionamentos

A estrutura mecânica do protótipo é constituída maioritariamente por perfis técnicos em liga de alumínio de modo a manter o seu peso o menor possível.

É composta por duas mesas iguais com 800x600 mm<sup>2</sup> cada, sendo a sua ligação feita por uma articulação do tipo *dumper* e funcionando como sistema de direção. Este sistema é formado por quatro chumaceiras interligadas por um eixo de articulação, de 35 mm de diâmetro, como é visível na Figura 2.1.

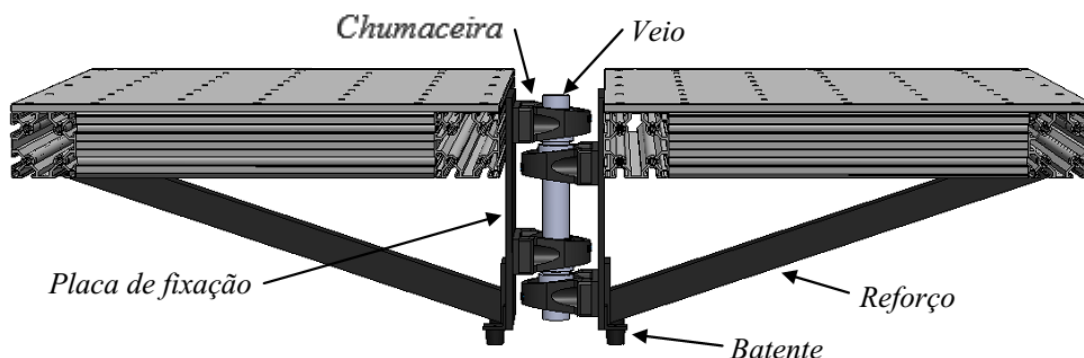


Figura 2.1 - Articulação de direção <sup>[1]</sup>

Este sistema é acionado por dois cilindros com um curso de 200 mm colocados em lados opostos das mesas, tal como se vê na Figura 2.2, permitindo um ângulo máximo de viragem de  $17^\circ$  para cada um dos lados.

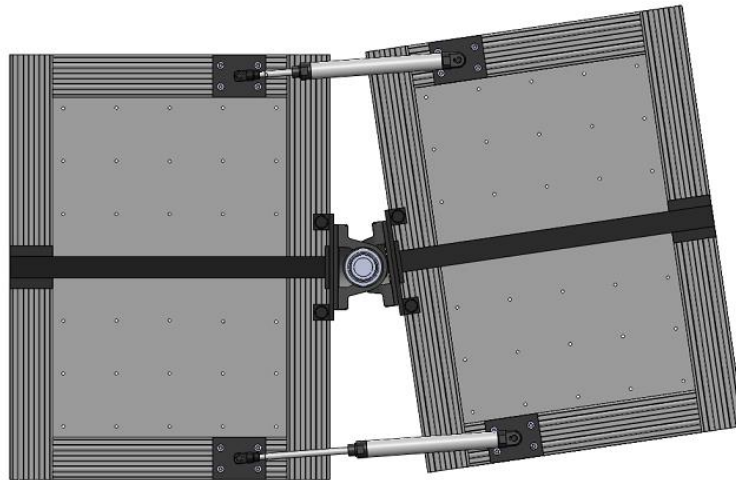


Figura 2.2 - Sistema de acionamento da direção <sup>[1]</sup>

Com estes valores é possível saber qual o raio de curvatura mínimo do AGV, tendo este sido avaliado em 5,85 m.

O veículo é ainda composto por quatro pernas iguais, tendo cada uma um cilindro de elevação e um cilindro de articulação com um curso de 200 mm e 100 mm respetivamente. No extremo de cada perna foram colocadas duas rodas movidas por um motor hidráulico. Cada atuador está assinalado no modelo 3D da Figura 2.3.



Figura 2.3 - Modelo 3D da perna da Mesa Hidráulica Móvel <sup>[1]</sup>

Os cilindros permitem a elevação e inclinação do veículo, sendo as rodas responsáveis pelo movimento de translação. Com recurso a estas pernas é possível subir e descer a superfície da mesa entre 600 mm e 1060 mm do nível do chão, como ilustrado na Figura 2.4. <sup>[1][2]</sup>

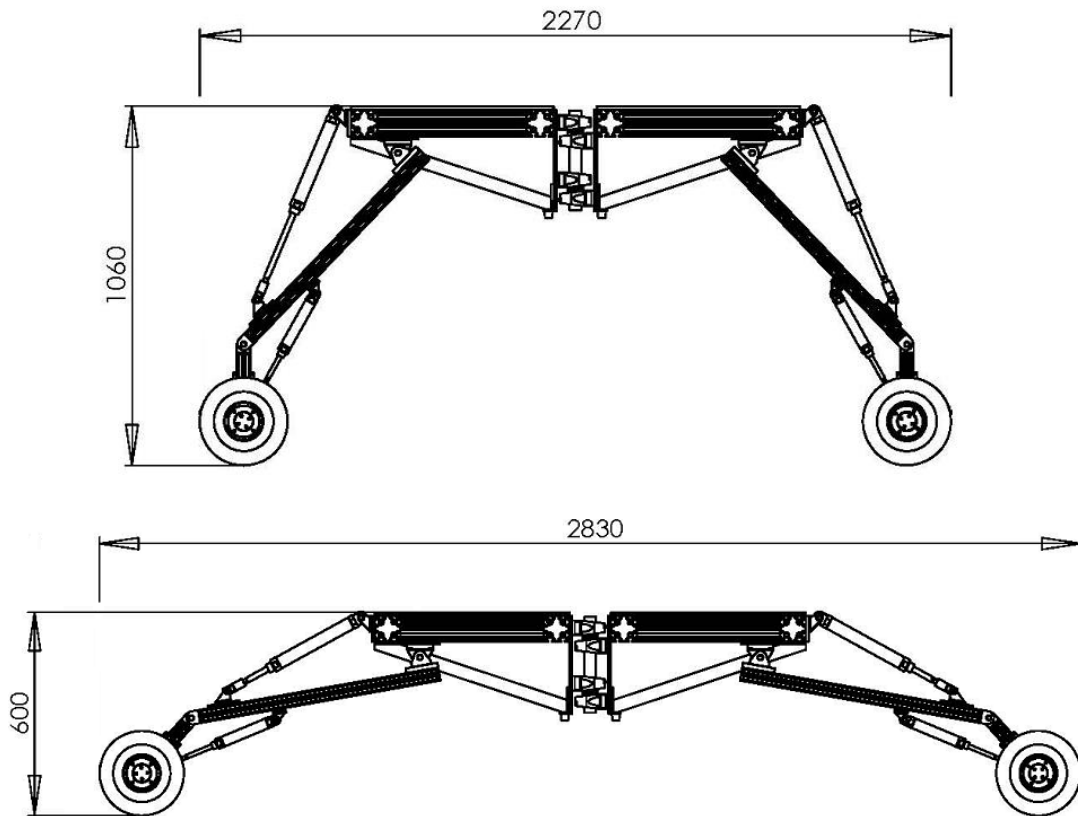


Figura 2.4 - Estrutura mecânica da Mesa Hidráulica Móvel <sup>[1]</sup>

## 2.2 Sistema hidráulico e fonte de energia

O sistema hidráulico está dividido em cinco sub-circuitos. Estes são alimentados por uma central hidráulica compacta, ver Figura 2.5, composta por um motor elétrico AC de 370 W, uma bomba de engrenagem exterior de 3,2 cm<sup>3</sup> de cilindrada e um reservatório cilíndrico. A bomba permite fornecer um caudal máximo de 4 l/min a uma pressão máxima de 35 bar.



Figura 2.5 – Exemplo de central hidráulica compacta <sup>[1]</sup>

De modo a permitir que a partir de uma fonte hidráulica de caudal se possam alimentar independentemente e simultaneamente 5 sub-circuitos, a alimentação de caudal é efetuada por um divisor de caudal de cinco elementos que disponibiliza um caudal máximo de 0,8 l/min para cada sub-circuito.

De forma a ser o mais compacto e leve possível, as válvulas de comando de cada sub-circuito hidráulico encontram-se agrupadas num bloco de liga de alumínio sendo as válvulas “de roscar”, para maior compacidade da solução.

Existem quatro blocos idênticos, cada um dedicado ao comando do movimento de uma das pernas do veículo, e um quinto para comandar a direção do mesmo.

Cada sub-circuito de acionamento de uma perna é constituído pelo comando do cilindro de elevação, do cilindro de articulação e do motor hidráulico, como é visível na Figura 2.6, no pressuposto de estes serem movimentos nunca simultâneos.

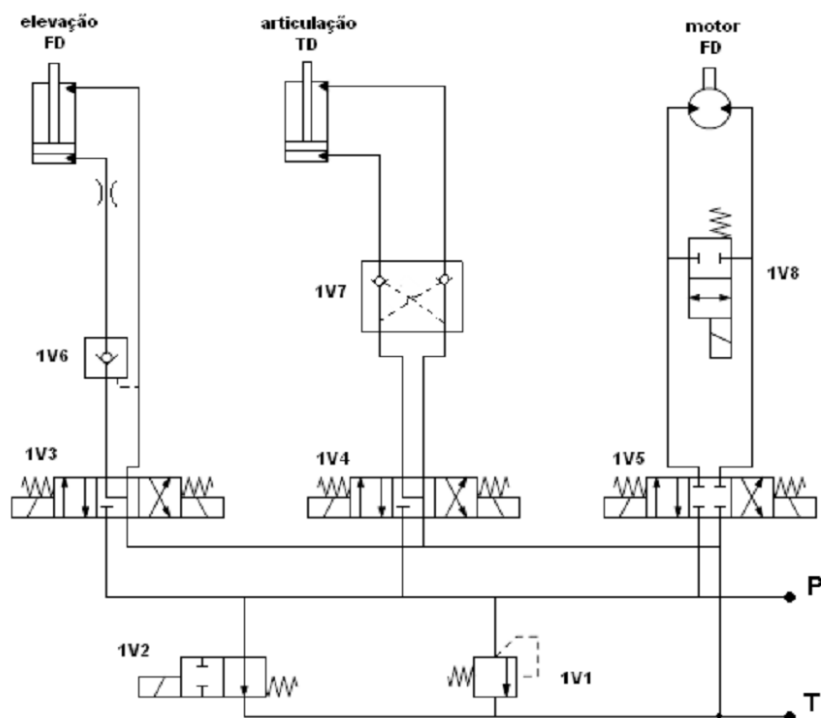


Figura 2.6 - Sub-circuito hidráulico de acionamento de uma perna <sup>[1]</sup>

Começando por analisar a elevação pode-se observar que o comando do seu avanço e recuo é realizado por uma eletroválvula 4/3 (1V3). Note-se ainda a existência de uma válvula de retenção pilotada simples (1V6) que garante que não existirá descida do AGV por efeito da gravidade.

O comando do cilindro de articulação é efetuado, de forma idêntica ao de elevação, usando uma electroválvula 4/3 (1V4). Porém utiliza-se uma válvula de retenção pilotada dupla (1V7) para impedir quer o avanço quer o recuo do cilindro por ação gravítica, uma vez que esta pode acontecer nos dois sentidos.

É de evidenciar que sendo cada bloco dedicado a uma perna, o comando da articulação corresponde efetivamente ao da perna oposta. Por exemplo, o bloco da perna da frente direita controla a articulação da perna de trás direita e vice-versa. Isto permite o comando simultâneo dos dois movimentos de uma mesma perna, sem interferência entre os dois movimentos.

Para o comando dos motores hidráulicos encontrados no extremo de cada perna recorre-se a uma eletroválvula 4/3 de centro fechado (1V5), o que permite uma rotação das rodas em ambos os sentidos, bem como a sua travagem aquando da não atuação da eletroválvula. Porém, existe a necessidade de ter rodas soltas para os movimentos de subida e descida da mesa, para atingir tal objetivo utiliza-se uma eletroválvula 2/2 (1V8) que permite destravar a roda.

Encontram-se ainda presentes, em cada sub-circuito, duas válvulas entre a linha de pressão e de retorno ao reservatório. Uma válvula limitadora de pressão (1V1) que impede a subida da pressão para valores perigosos e uma eletroválvula 2/2 (1V2) que quando atuada coloca o respetivo sub-circuito em carga, o que é útil quando se pretende que não existam movimentos de uma perna específica.

Existe um sexto bloco que se encontra à saída para o reservatório destes quatro primeiros sub-circuitos e é constituído por duas válvulas. Este sub-circuito, designado “de contrapressão”, está representado na Figura 2.7 e tem como função impor, ou não, uma contrapressão na linha de retorno quando esta é precisa para os movimentos de descida. Para tal recorre-se a uma válvula limitadora de pressão (6V1), para definir o valor da contrapressão, e uma eletroválvula 2/2 (6V2) que deixa “curto-circuitar” a função da primeira.

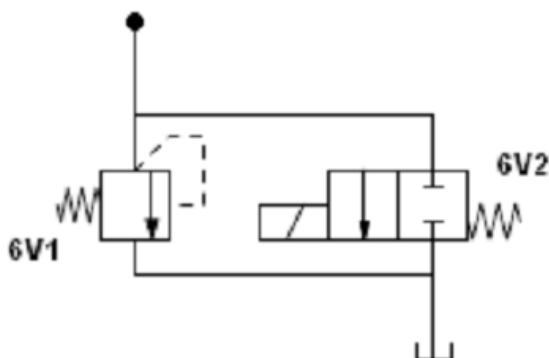


Figura 2.7 - Sub-circuito de contrapressão <sup>[1]</sup>

O último sub-circuito presente no circuito hidráulico do protótipo é o da direção e encontra-se simbolicamente mostrado na Figura 2.8. Sendo composto por uma válvula limitadora de pressão (5V1) e um eletroválvula 4/3 (5V2) para comando da direção do AGV, esta tem a posição central a realizar a descarga do fluido. Este sub-circuito tem a particularidade de ter as ligações cruzadas para os dois cilindros de direção, o que poderia levar a problemas de sincronismo uma vez que estes são assimétricos. Porém este sincronismo é garantido mecanicamente visto que os atuadores são alimentados em paralelo e os esforços realizados são devidamente invertidos.

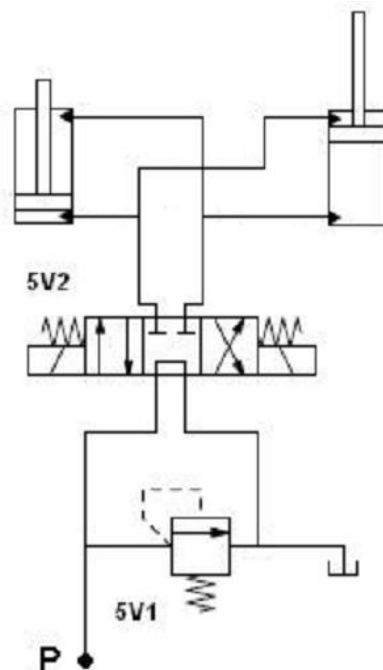


Figura 2.8 - Sub-circuito de direção <sup>[1]</sup>

Por fim, é de evidenciar que qualquer movimento feito pelos atuadores tem a possibilidade de ser efetuado a diferentes velocidades. Isto deve-se à incorporação de um variador de frequência no veículo, o que permite controlar a velocidade do motor elétrico da fonte de

energia e, consequentemente, o caudal debitado pela bomba hidráulica numa ampla gama de velocidades.

O aspeto final do sistema hidráulico aplicado na Mesa Hidráulica Móvel é visível na Figura 2.9. <sup>[1][2]</sup>

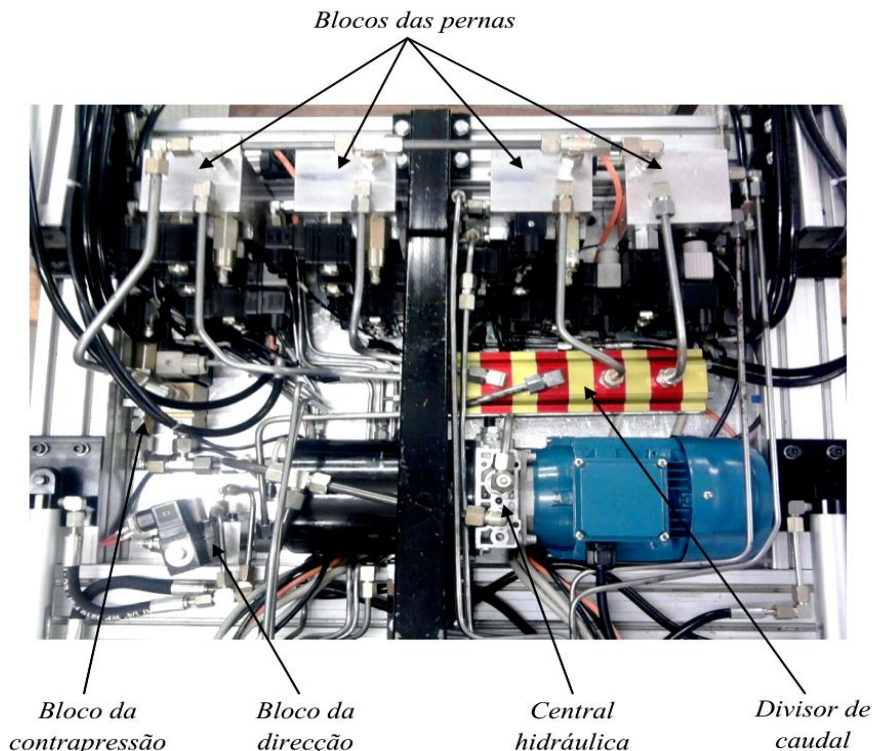


Figura 2.9 - Sistema hidráulico da Mesa Hidráulica Móvel <sup>[1]</sup>

### 2.3 Sensorização

O AGV encontra-se equipado com diversos sensores e transdutores para monitorizar o seu estado no decorrer do seu funcionamento. Serão agora descritos cada um desses componentes.

O transdutor presente em maior número (oito) são transdutores de cabo potenciométricos estando aplicados em cada um dos atuadores que fazem a elevação e a articulação. Estes transdutores de posição são da marca *Micro-Epsilon*, mais especificamente, do modelo *MK30* (Figura 2.10) e têm uma gama de medição de 250 mm e uma resolução de 0,1 mm.



Figura 2.10 - Transdutor de posição *Micro-Epsilon MK30* <sup>[1]</sup>

Foi usado outro transdutor de posição potenciométrico, este de haste, que tem como propósito medir indiretamente o ângulo de direcção. O transdutor aplicado para este fim é o modelo *LTH100S* da marca *GEFRAN* (Figura 2.11) e tem um curso de 100 mm. Repare-se que

os cilindros hidráulicos usados para o comando da direção tendo um curso de 200 mm, obrigam à instalação do transdutor de posição num local mais próximo do eixo de rotação.

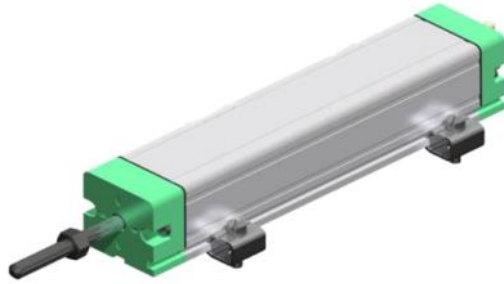


Figura 2.11 - Transdutor de posição *LTH100S* <sup>[1]</sup>

Com o objetivo de saber se alguma perna se encontra em suspenso foram colocados transdutores de pressão. Estes foram aplicados nas linhas de alimentação da câmara principal de cada cilindro de elevação da mesa, as câmaras que sentem a carga sobre cada perna. Optou-se por transdutores de pressão de membrana do modelo *TP3BE53R* da marca *AEP Transducers*, que apresentam uma gama de medição de 0 a 50 bar e se encontram representados na Figura 2.12.



Figura 2.12 - Transdutor de pressão *TP3BE53R* <sup>[1]</sup>

O sensor mais específico para este veículo é, provavelmente, o inclinómetro, pois é esse que vai permitir aferir a horizontalidade do mesmo. Necessitava-se de um inclinómetro de dois eixos ortogonais e optou-se pelo modelo *SCA121T* da *VTI Technologies*, com uma gama de medição entre -30° e 30° em ambos os eixos excedendo o requerido pelo protótipo. Com a instalação deste componente, exibido na Figura 2.13, é possível ler a inclinação transversal e lateral do veículo com uma resolução de 0,1°.



Figura 2.13 - Inclinómetro *SCA121T* <sup>[2]</sup>

Por fim, foram aplicados sensores óticos para possibilitar a deteção de um percurso marcado no solo. Com esse fim em vista foram selecionados os sensores fotoelétricos *XUB0BPSNM12* da *Schneider Electric* que quando colocado no modo de funcionamento de proximidade é capaz de detetar a fita preta a distâncias entre 4 e 13 cm. Colocaram-se dois em cada perna, dando um total de oito sensores óticos, mostrados na Figura 2.14. <sup>[1][2]</sup>





Figura 2.14 - Sensor ótico XUB0BPSNM12 [2]

## 2.4 Sistema elétrico e hardware

O cérebro da Mesa Hidráulica Móvel é o seu autômato programável pois é este que lê todos os sensores e atua todas as eletroválvulas. O autômato aplicado na mesa foi um micro autômato programável de características industriais, o *Twido* da *Schneider Electric*, a sua representação e a dos módulos de expansão utilizados pode ser vista na Figura 2.15.

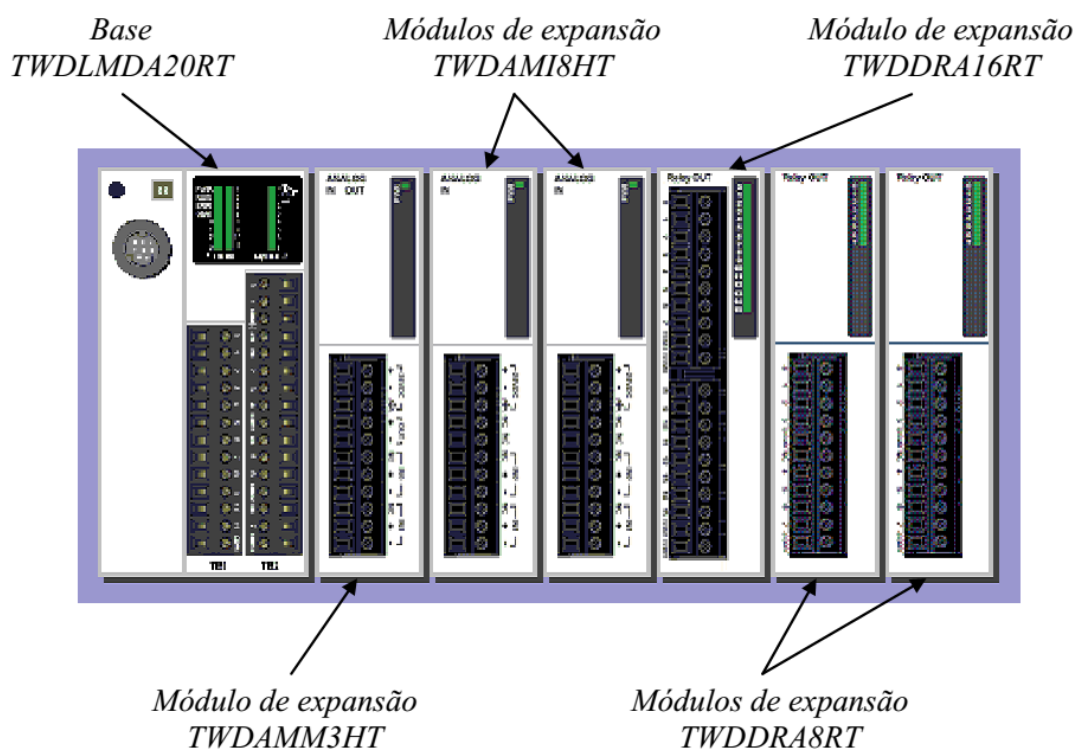


Figura 2.15 - Autômato e módulos de expansão [1]

O autômato usado é composto por um módulo base e outros seis módulos de expansão, reunindo as seguintes especificações.

- Uma base modular *TWDLMDA20DRT*, que possui o processador que corre o programa de controlo do AGV, doze entradas e oito saídas digitais (seis por relé);
- Um módulo de expansão *TWDAMM3HT*, com uma saída e duas entradas analógicas;
- Dois módulos de expansão *TWDAMI8HT*, com oito entradas analógicas cada;
- Um módulo de expansão *TWDDRA16RT*, com dezasseis saídas digitais (por relé);
- Dois módulos de expansão *TWDDRA8RT*, com oito saídas digitais cada (por relé).



Recorrendo à Tabela 1 podem-se observar quais as funções desempenhadas por cada módulo do autômato.

Tabela 1 - Funções dos diversos módulos do autômato

Módulo	Funções
<i>TWDLMDA20DRT</i>	<ul style="list-style-type: none"> <li>• Monitorização do botão de emergência</li> <li>• Atuação dos solenoides da direção</li> <li>• Atuação dos solenoides da contrapressão</li> <li>• Atuação do <i>start/stop</i> do variador de frequência</li> </ul>
<i>TWDAMM3HT</i>	<ul style="list-style-type: none"> <li>• Sinal de referência para o variador de frequência</li> <li>• Leitura do inclinómetro</li> </ul>
<i>TWDAMI8HT</i>	<ul style="list-style-type: none"> <li>• Leitura dos transdutores de posição</li> </ul>
<i>TWDAMI8HT</i>	<ul style="list-style-type: none"> <li>• Leitura da carga das baterias</li> <li>• Leitura do transdutor da direção</li> <li>• Leitura dos transdutores de pressão</li> </ul>
<i>TWDDRA16RT</i>	<ul style="list-style-type: none"> <li>• Atuação dos solenoides da perna frente direita</li> <li>• Atuação dos solenoides da perna frente esquerda</li> </ul>
<i>TWDDRA8RT</i>	<ul style="list-style-type: none"> <li>• Atuação dos solenoides da perna trás direita</li> </ul>
<i>TWDDRA8RT</i>	<ul style="list-style-type: none"> <li>• Atuação dos solenoides da perna trás esquerda</li> </ul>

Para se poder variar a velocidade dos atuadores foi instalado um variador de frequência que controla a velocidade do motor elétrico e, consequentemente, o caudal da bomba hidráulica. O modelo instalado no protótipo foi o *Varispeed-606PC3* da *Yaskawa Electric Corporation* (Figura 2.16) que é comandado por um sinal de referência enviado pelo autômato. Este componente requer uma alimentação de 240 VAC monofásicos e fornece à saída 240VAC trifásicos, sendo capaz de variar a frequência entre 0 e 60 Hz.



Figura 2.16 - Variador de frequência *Varispeed-606PC3*<sup>[1]</sup>

Como a alimentação da Mesa Móvel é feita através de baterias de 12 VDC, há uma necessidade de converter esta energia em corrente alternada 240 VAC. Esta função é desempenhada pelo inversor de corrente *PSI100S* da *HQ-Power*, exibido na Figura 2.17.<sup>[3]</sup>



Figura 2.17 - Inversor de corrente *PSI100S*<sup>[3]</sup>

Embora o esquema elétrico de alimentação do veículo seja algo complexo é possível ser dividido em quatro sub-sistemas independentes. Estes são:

- *24 V de comando*, circuito alimentado por duas baterias de 12 VDC ligadas em série, cuja responsabilidade é alimentar o autômato e os transdutores de pressão;
- *10 V de sinal*, este circuito é criado a partir duma fonte de alimentação que é alimentada pelas baterias acima referidas e gera uma tensão constante de 10 VDC para os transdutores de posição, de modo a que as leituras não sejam influenciadas pelas variações de carga das baterias;
- *24 V de potência*, responsável por alimentar os solenoides das eletroválvulas, este circuito é gerado por duas baterias de 12 VDC em série;
- *12 V de potência*, este circuito é o que exige mais corrente e é composto por quatro baterias de 12 VDC ligadas em paralelo. Serve para alimentar o inversor de corrente que, por sua vez, gera os 230 VAC monofásicos que são convertidos em trifásicos pelo variador de frequência que alimenta o motor elétrico da central hidráulica.

Para seccionar globalmente estes circuitos é usada um interruptor de painel *on/off* que corta toda a alimentação do AGV. São ainda usadas duas botoneiras de emergência que desligam todos os sub-sistemas elétricos à exceção do de 24 V de comando que alimenta o autômato, colocando o veículo numa posição de segurança em que está ligado, mas incapaz de realizar qualquer movimento. O esquema que representa este sistema de alimentação pode ser observado na Figura 2.18.<sup>[1][2]</sup>

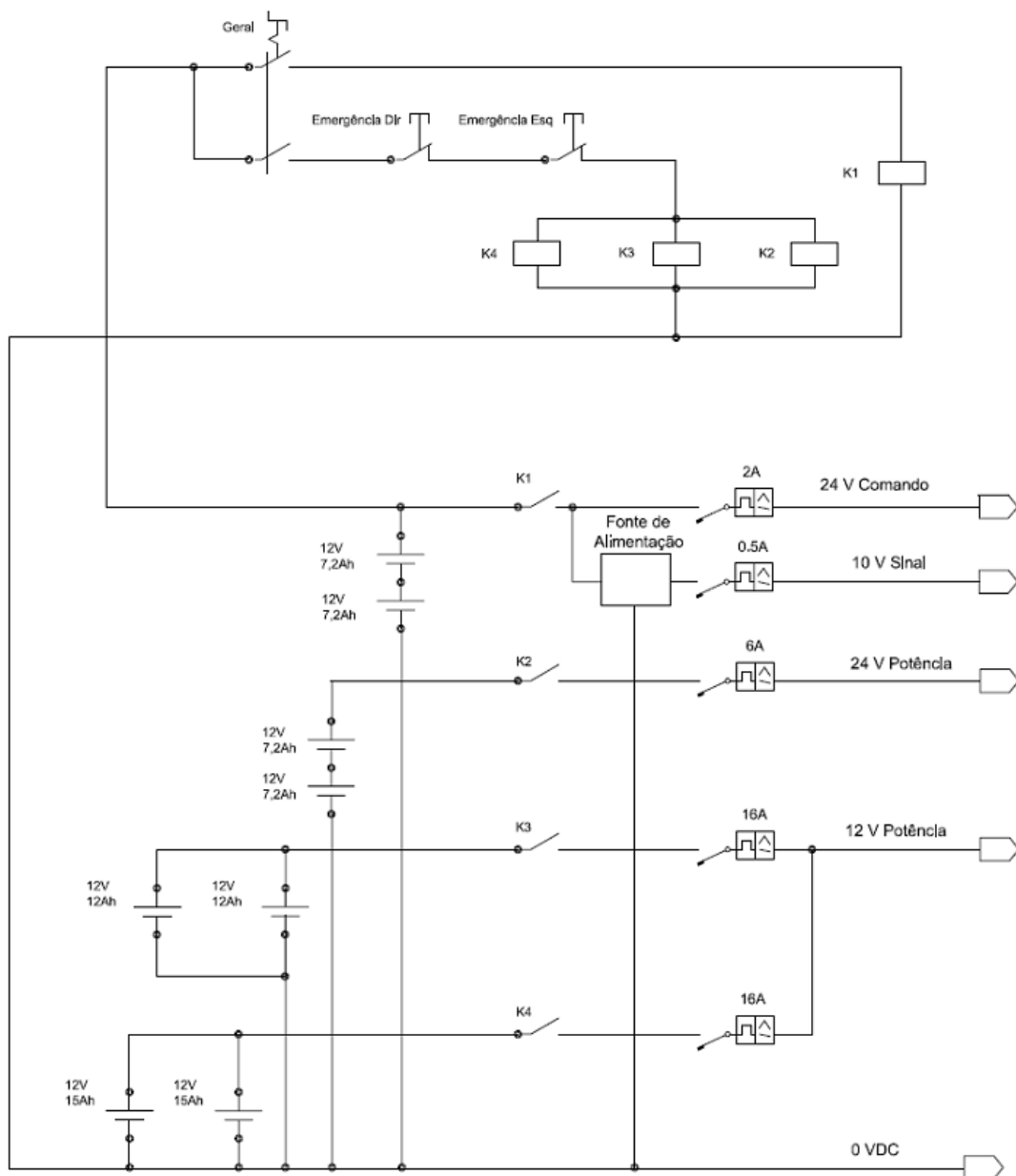


Figura 2.18 - Esquema de alimentação elétrica <sup>[2]</sup>



### 3 Análise de sistemas de navegação

Neste capítulo pretendem-se analisar quais os métodos de guiamento de AGVs mais usados atualmente, para desse modo se poder decidir que método usar para substituir a solução instalada no presente protótipo, que envolve sensores óticos.

Será feita uma análise crítica de cada sistema revelando os pontos fortes e fracos de cada solução bem como a sua adequabilidade para o caso vertente.

Por fim, será justificada a opção selecionada para implementar nesta nova fase da Mesa Hidráulica Móvel.

#### 3.1 Sistemas de navegação

Qualquer veículo que pretenda mover-se de modo autónomo tem de ser equipado com um sistema de navegação que lhe permita determinar qual o próximo movimento a executar. Os veículos com tal capacidade são denominados AGV's.

Existem múltiplas soluções para implementar um sistema de navegação autónomo, sendo que a decisão por qual se optar recai nos requisitos de flexibilidade, custos de aplicação e ambiente de trabalho.

Neste momento existem AGVs de muitas formas e para inúmeros fins, mas o seu modo de navegação pode ser incluído num dos seguintes grupos.

##### 3.1.1 Filoguiado

Esta solução é baseada na colocação de um cabo elétrico embutido no solo, a cerca de 3 cm da superfície, e dotando o veículo de um sensor eletromagnético que é capaz de detetar o campo magnético criado pela corrente que passa no cabo elétrico.

O sistema filoguiado é ideal para instalações que pretendem ter um percurso fixo com elevada precisão que se mantenha operacional por um longo tempo, uma vez que é um sistema simples e robusto para tal fim.

Contudo, qualquer necessidade de alteração do percurso ou reparação terá custos elevados uma vez que é necessário abrir e refazer o pavimento. A esta desvantagem associa-se o facto de poderem existir interferências causadas por cabos elétricos que se possam encontrar nas proximidades.<sup>[2][4]</sup>

Na Figura 3.1 pode-se ver um exemplo deste tipo de solução.



Figura 3.1 - AGV com navegação por filoguiamento <sup>[2]</sup>

### 3.1.2 Sensorização ótica

Uma solução muito similar à anterior, baseada igualmente num percurso fixo, é realizada trocando apenas o cabo elétrico por uma fita de cor (adesiva ou pintada) e o sensor magnético por um sensor ótico, como o exemplo mostrado na Figura 3.2.

Este é o sistema que está presente de momento na mesa hidráulica em estudo e apresenta a vantagem, relativamente ao filoguiado, de permitir alterações do percurso e reparações com uma maior facilidade. No entanto, existe maior probabilidade de erro devido à fita ficar suja ou danificada, o que se traduz numa menor fiabilidade. <sup>[2][4]</sup>



Figura 3.2 - AGV com navegação por sensores óticos <sup>[2]</sup>

### 3.1.3 Grelha de *transponders*

Este sistema permite uma maior flexibilidade do que os anteriores, pois não necessita de um percurso fixo. Funciona através da colocação de uma grelha de *transponders* embutidos no chão que servem de referência para o AGV.

Equipado com um giroscópio e um sensor de *Hall* o veículo vai avançando de marcador em marcador, que eliminam o possível desvio associado ao erro do giroscópio, cobrindo um percurso que é facilmente alterado por via de programação.

Os valores de espaçamento entre os marcadores variam de instalação para instalação, mas ronda o valor de 7,5 metros. <sup>[2][4][5]</sup>

Veja-se na Figura 3.3 uma representação esquemática deste tipo de navegação.

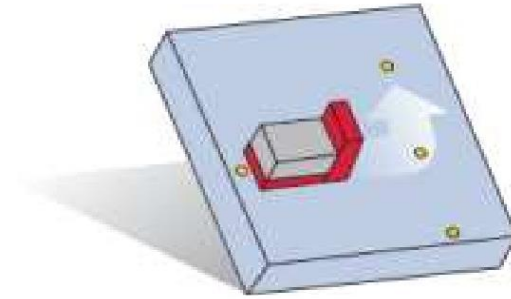


Figura 3.3 - Esquema de AGV com navegação por grelha de *transponders* <sup>[2]</sup>

### 3.1.4 Triangulação por laser

O veículo vai efetuando varrimentos rotativos analisando o ambiente circundante com um *scanner laser*. Este deteta a distância a que se encontra de cada painel refletor colocado em parede ou coluna com coordenadas conhecidas. Sabendo isto o AGV é capaz de calcular a sua posição através do método da triangulação, sendo mais exato quanto maior o número de sensores encontrados.

Esta solução, cujo esquema é observável na Figura 3.4, permite uma elevada flexibilidade, sendo o veículo capaz de navegar por qualquer percurso, desde que mantenha contacto visual com um mínimo de três refletores. Utilizando este método é-se capaz de elevada precisão. No entanto, a existência de excessiva poeira no ar ou uma obstrução na linha de visão entre o *scanner* e os refletores poderá leva à falha do sistema. <sup>[2][4][5]</sup>

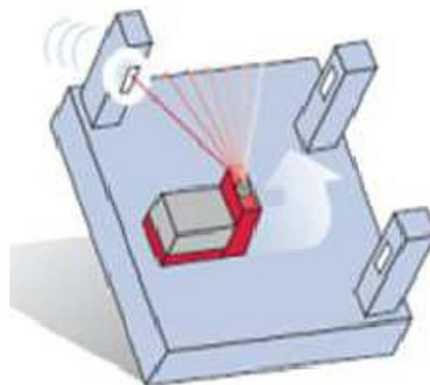


Figura 3.4 - Esquema de AGV com navegação por triangulação <sup>[2]</sup>

### 3.1.5 Visão artificial

A navegação com recurso a visão é, provavelmente, a que mais potencialidades futuras apresenta de entre todas. No entanto, é também a que apresenta mais desafios.

Baseada no modo de navegação dos humanos e de muitos outros animais, este sistema funciona através do reconhecimento da posição e/ou alvo utilizando informação visual.

Atualmente é das soluções em maior crescimento e estudo, tendo sido descobertas algumas formas de a usar na navegação. Pode ser utilizada num modo similar ao anterior, triangulando a sua posição através de marcos de geometria conhecida, ou seguindo um alvo, retirando a distância para o mesmo da proporção entre a imagem aparente e o seu tamanho real conhecido.

Apresenta capacidades únicas como a deteção de cores e identificação de objetos, sendo capaz de distinguir uma pessoa de outro obstáculo. Contudo ainda é bastante sensível a

variações na luminosidade e pouco fiável, razão pela qual tem vindo a ser apenas usada em conjunto com outras tecnologias. <sup>[2]</sup>

### 3.1.6 Localização GPS

A localização por GPS é apenas utilizada em situações de navegação em ambientes exteriores e de grande dimensão. Isto deve-se à falta de precisão que este sistema ainda revela em comparação com todos os outros.

Recorrendo a um método de triangulação por satélites, este sistema é muito útil para grandes distâncias onde a instalação de outra opção se revelaria muito dispendiosa. Mas esta vantagem tem o custo da perda de precisão da posição.

Esta tecnologia tem vindo a ser usada em transporte de mercadorias em ambiente portuário e até no transporte de passageiros em aeroportos.

É prática comum utilizar em conjunto com qualquer um destes modos de guiamento sistemas de segurança capazes de detetar obstáculos que se coloquem inesperadamente no percurso do AGV. São normalmente utilizados *bumpers* mecânicos ou sensores de proximidade para este tipo de função de segurança. <sup>[2]</sup>

## 3.2 Opção escolhida

O primeiro fator a ter em conta é o local de trabalho do nosso protótipo, que será o Laboratório de Óleo-hidráulica do edifício L (L-101) da FEUP. Uma vez que estão excluídas quaisquer opções que envolvam obras de construção civil e que nos encontramos num espaço pequeno e interior, as únicas alternativas que são opção são: a sensorização ótica, a triangulação por *laser* e a visão artificial.

Tendo sido definido como objetivo a evolução do modo de navegação podemos retirar o modo de sensorização ótica visto ser o utilizado presentemente na Mesa Hidráulica Móvel.

Restam como opção a triangulação por *laser* ou visão artificial. O local de trabalho do veículo tem um elevado número de equipamentos e está sujeito a constantes alterações na sua disposição, o que torna aparentemente impossível a colocação de marcadores que fiquem visíveis a todo o momento. Por este motivo, e pelo elevado custo de um *scanner laser*, tomou-se como opção utilizar um modo de navegação baseado em visão artificial.

Como já foi referido será impraticável a colocação de marcadores, razão pela qual se pretende que o sistema de guiamento a usar seja o de seguimento de um alvo, que no caso presente será seguir um alvo que poderá ser um colete refletor.

A principal preocupação associada a este tipo de solução reside na variação de luminosidade e na obstrução da linha de visão. Embora o laboratório possua janelas, as suas dimensões são reduzidas e a principal fonte de iluminação é artificial, razão pela qual se anula a primeira preocupação. Quanto à segunda, esta será suprimida pela incorporação de sensores de proximidade que fornecerão uma segurança adicional e um modo de deteção de obstáculos ao protótipo.

Esta solução permite uma enorme flexibilidade, pois o percurso é definido pelo operador (com o colete vestido) no momento e apenas restringido pelo baixo ângulo de viragem da mesa. Uma vez que esta é simétrica, decidiu-se colocar uma câmara e sensores de proximidade em ambos os lados do protótipo, permitindo um funcionamento igual em ambas as direções.



## 4 Conceção e desenvolvimento da nova solução

Terminada a análise ao estado atual do protótipo, e sabendo quais os pontos a melhorar, vão ser expostos neste capítulo quais os componentes e funcionalidades que foram aplicados para cumprir esses pontos.

Para melhor se entenderem todas as alterações efetuadas, o presente capítulo foi dividido da seguinte forma:

- *Visão artificial*, exposição da câmara escolhida bem como o sistema para processamento de imagem;
- *Sensores de proximidade*, explicação da escolha do modelo, modo de funcionamento e esquema elétrico;
- *Suporte para sensores e câmara*, apresentação da conceção e desenho dos suportes;
- *Sistema elétrico e Hardware*, descrição das alterações efetuadas no sistema elétrico e instalação de um novo autómato programável.

Todo este capítulo está focado no *hardware* aplicado à Mesa Hidráulica Móvel, sendo que a sua programação é abordada no capítulo seguinte.

### 4.1 Visão Artificial

Como explicado no capítulo anterior, a opção sobre o modo de navegação recaiu na visão artificial.

Uma vez que o maior foco do orçamento foi colocado na aquisição de um PLC (*Programmable Logic Controller*) com boa capacidade e de baterias sem as quais seria impossível mover o AGV, o orçamento para o restante *hardware* ficou limitado. Tendo isso em conta a maioria das opções utilizadas em ambiente industrial ficaram fora do alcance, tendo-se optado por uma solução muito comum em protótipos e projetos mais demonstrativos, tal como o presente. Esta é a utilização de um microcomputador para efetuar o processamento de imagem e a utilização de uma câmara que seja compatível com o mesmo.

Após alguma pesquisa e comparação foi decidido como caminho a seguir a aquisição de duas placas de microcomputador *Raspberry Pi 3* modelo B (Figura 4.1), que é um microcomputador de baixo custo, bastante utilizado neste tipo de projetos e cujas características cumprem os requisitos da mesa.



Figura 4.1 - *Raspberry Pi 3* modelo B <sup>[6]</sup>

Este componente, embora seja do tamanho de um cartão de crédito, possui um processador *QuadCore* de 1,2GHz e 1GB de RAM. Possui múltiplas entradas USB, uma entrada *Ethernet* (que será usada para comunicar com o PLC), conectores GPIO (*General Purpose Input Output*), uma ficha para câmara CSI e comunicação por *Wi-Fi* e *Bluetooth*.

O processador é capaz de executar vários programas em simultâneo e de correr programas complexos. Porém não é capaz de fazer processamento de imagem em tempo real. Embora seja algo normalmente exigido para uma aplicação destas, a baixa dinâmica e velocidade demonstrada pelo veículo de estudo permitem que esta solução seja aplicável para o caso presente.

A alimentação do microcomputador é feita através de um cabo mini USB com 5V, igual à da maioria dos telemóveis, tendo sido utilizada a alimentação de 230 VAC fornecida pelo inversor de corrente e utilizados dois carregadores comuns de telemóvel para alimentar as duas *Raspberry Pi*. <sup>[7]</sup>

Com o microprocessador definido é necessário seleccionar qual o componente que vai captar a informação visual. Embora hajam múltiplas escolhas válidas, a decisão incidiu na *Raspberry Pi Camera Module* uma câmara de alta definição (resolução de 3280 por 2464) de dimensões reduzidas (23,86 por 25 mm), com 8 megapixel e uma taxa de transferência máxima de imagem de 30 fps a 1080p. As características mais atrativas neste componente, mostrado na Figura 4.2, para além da sua boa qualidade de imagem, são o seu controlo automático à exposição solar, que ajuda a combater as questões de flutuação de luminosidade, e compatibilidade com a *Raspberry Pi*, ligando-se a esta pela ficha CSI. <sup>[8]</sup>

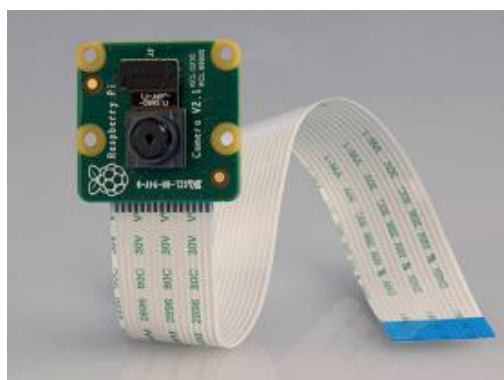


Figura 4.2 - *Raspberry Pi Camera Module* <sup>[8]</sup>

Esta solução apresenta uma excelente compatibilidade entre si, visto terem sido criados para funcionarem juntos, uma capacidade de aquisição e processamento de imagem baixo para ser usado em sistemas de tempo real, mas suficiente para o problema em estudo, um baixo custo e dimensões reduzidas, o que é ideal devido ao limitado espaço disponível debaixo das mesas do veículo.

Foi adquirido um par de cada um destes componentes, pois pretende-se instalar um sistema simétrico que permita um funcionamento idêntico nos dois sentidos do protótipo.

O papel destes componentes é a deteção de um colete refletor, sendo que se entrará em mais detalhe no capítulo seguinte.

## 4.2 Sensores de proximidade

A utilização destes sensores é essencial, especialmente para fins de segurança, permitindo a deteção de objetos que se encontrem no percurso da Mesa Móvel e forçando-a a parar. Contudo será vantajoso utilizar os sensores de proximidade, não como detetores, mas como transdutores e criar uma grelha com escalonamento de distâncias, similar à da Figura 4.3, que permita que o veículo se desvie de alguns obstáculos.

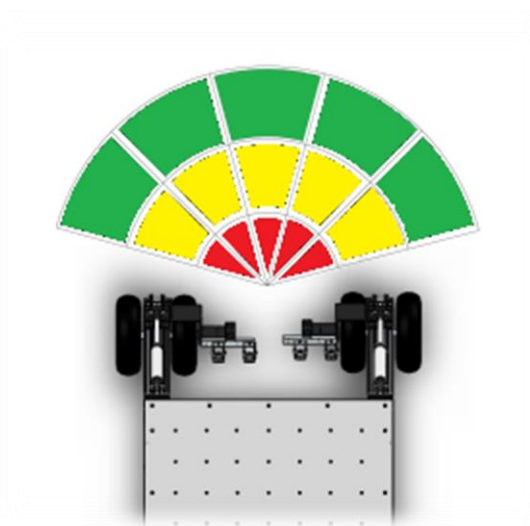


Figura 4.3 - Grelha de deteção de proximidade

A oferta a nível de sensores de proximidade é vasta, mas colocando a restrição de ser um transdutor e não simplesmente um detetor, diminui-se o leque de escolhas. Correspondentes a estas características, as opções que sobressaem são os sensores laser e os ultrassónicos, sendo que entre estes dois se optou pelos últimos devido ao seu preço mais apelativo e compatibilidade.

A escolha recaiu num sensor de proximidade ultrassónico, de baixo custo e dimensão, que normalmente é utilizado em projetos que envolvam *Raspberry Pi* devido à sua boa compatibilidade. Optou-se pelos sensores HC-SR04, que podem ser vistos na Figura 4.4, que são alimentados a 5 V e detetam objetos entre 2 cm e 4 m de distância e abrangem um ângulo de  $\pm 15^\circ$ .



Figura 4.4 - Sensor de proximidade ultrassônico HC-SR04 <sup>[9]</sup>

Este modelo possui quatro pinos de ligação sendo eles a tensão de alimentação (Vcc) pela qual o sensor é alimentado, o *trigger* que despoleta o impulso ultrassônico, o *echo* que devolve um sinal proporcional à distância do objeto e o *ground* (GND) que fecha o circuito.

O seu funcionamento baseia-se na emissão de um sinal sonoro e a captação da sua reflexão num objeto. Para este componente em concreto o seu funcionamento encontra-se exposto no diagrama temporal da Figura 4.5. <sup>[9]</sup>

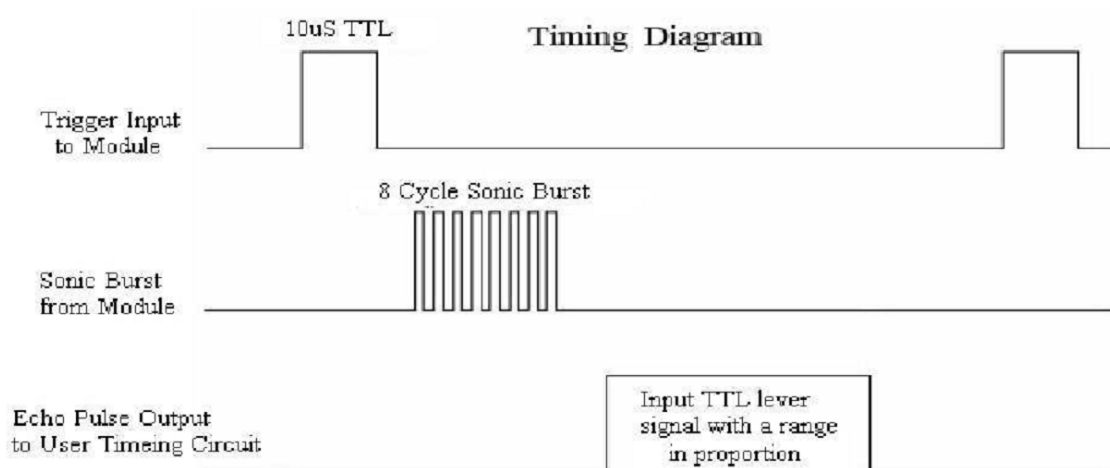


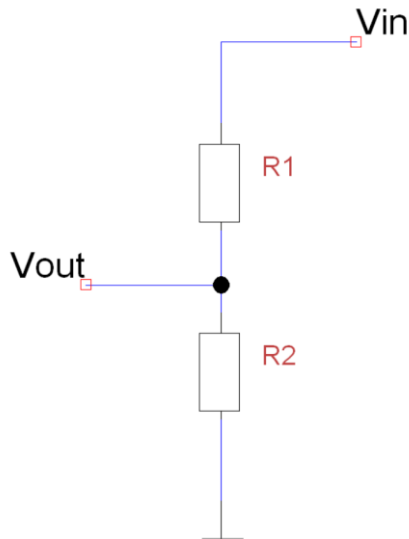
Figura 4.5 - Diagrama temporal de funcionamento do HC-SR04 <sup>[9]</sup>

É de notar que aquando da emissão de um impulso de 10 µs o sensor envia um sinal de oito ciclos seguidos de 40 kHz e leva o valor da linha de *echo* a *high*. Quando deteta um eco na mesma frequência leva novamente a linha *echo* a *low*, permitindo retirar pela equação abaixo o valor da distância do objeto detetado, uma vez que o tempo permanecido em *high* corresponde ao tempo demorado pela velocidade do som (343 m/s) a ir de encontro com o objeto e regressar.

$$distância [m] = \frac{tempo\ em\ high [s] \times 343 [m/s]}{2}$$

Note-se que, como já referido anteriormente, cada sensor só é capaz de cobrir um ângulo total de 30° o que gera uma necessidade de ter múltiplos sensores em cada lado da mesa para permitir um ângulo de deteção maior. No total foram colocados 5 sensores de cada lado, obtendo 150° de deteção em cada um dos extremos do protótipo.

Utilizando os pinos GPIO, cada *Raspberry Pi* é capaz de alimentar e comandar os cinco sensores em sequência, sendo, no entanto, necessária a colocação de um divisor de tensão à saída da linha de *echo* de cada sensor, uma vez que estes produzem um sinal de 5 V e os *inputs* do microcomputador são de 3,3 V. O esquema de um circuito divisor de tensão é o da Figura 4.6 e o valor da relação entre resistência pode ser retirado pela equação abaixo.



$$\frac{V_{out}}{R_2} = \frac{V_{in}}{R_1 + R_2} \Leftrightarrow$$

$$\frac{V_{out}}{V_{in}} = \frac{R_2}{R_1 + R_2} \Leftrightarrow$$

$$\frac{3,3 [V]}{5 [V]} = \frac{R_2}{R_1 + R_2} \Leftrightarrow$$

$$0,66 \times R_1 + 0,66 \times R_2 = R_2 \Leftrightarrow$$

$$0,66 \times R_1 = 0,34 \times R_2 \Leftrightarrow$$

$$\frac{R_2}{R_1} = 1,94$$

Figura 4.6 - Circuito divisor de tensão

A combinação de resistências que mais perto fica da relação requerida e reduz o consumo de corrente foram as resistências de 3,3 kΩ e 6,8 kΩ, obtendo-se uma tensão de saída de 3,36 V e um consumo total para os cinco sensores de 2,5 mA.

Sabendo isto é possível ligar com segurança a *Raspberry Pi* a cada sensor HC-SR04 seguindo o esquema elétrico visível na Figura 4.7.

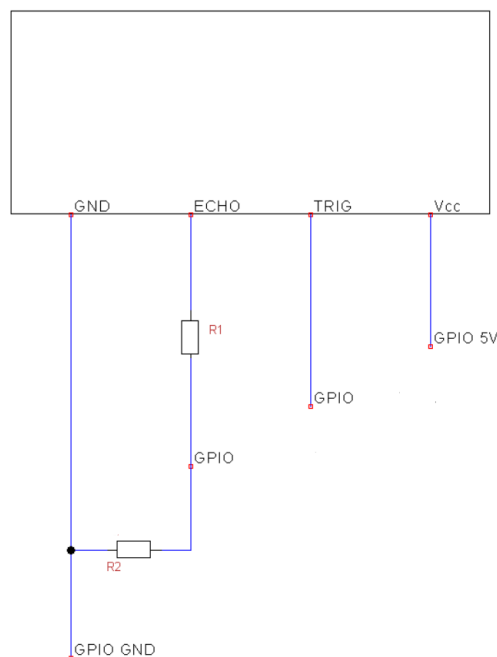


Figura 4.7 - Circuito de ligação do sensor ao microcomputador

Os pinos GPIO da *Raspberry Pi* têm funções diferentes, sendo importante saber a quais ligar os sensores. De seguida apresenta-se a Tabela 2 e a Figura 4.8 que permitem estabelecer as ligações corretas a efetuar entre os componentes do mesmo lado do protótipo (repetir de forma igual para o lado oposto). Note-se que aliado a esta operação é necessário referenciar qual o uso dos pinos GPIO na programação sendo esse assunto tratado no capítulo seguinte.

Tabela 2 - Ligações entre *Raspberry Pi* e HC-SR04

HC-SR04		Raspberry Pi (GPIO#)
Vcc (comum a todos os sensores)		2
GND (comum a todos os sensores)		6
Sensor 1	TRIG	5
	ECHO	23
Sensor 2	TRIG	6
	ECHO	24
Sensor 3	TRIG	13
	ECHO	25
Sensor 4	TRIG	19
	ECHO	12
Sensor 5	TRIG	26
	ECHO	16

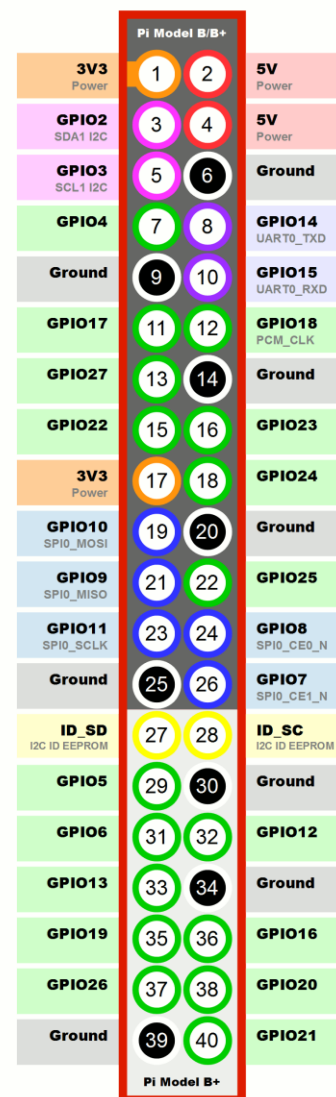


Figura 4.8 - Lista GPIO <sup>[10]</sup>

Com a instalação destes componentes torna-se possível a identificação de objetos que se intrometam no percurso da Mesa Hidráulica Móvel.

### 4.3 Suportes para sensores e câmaras

Na solução anterior existiu a necessidade de desenvolver um suporte que mantivesse permanentemente numa posição vertical os detetores óticos. No caso presente a posição ideal é a horizontal, mas o valor de inclinação máximo conseguido pela mesa não influencia as leituras em valores relevantes, razão pela qual se optou por uma estrutura fixa à mesma.

O principal requisito para o desenvolvimento deste suporte foi a criação de uma estrutura que alinhasse os cinco sensores com ângulos de 30° entre si e a sua colocação num local onde estes não detetassem as pernas da mesa.

O design encontrado foi o de uma chapa furada com 13 furos, um em cada extremo para fixação, cinco conjuntos de dois furos para o emissor e o recetor de cada sensor HC-SR04 e um furo mais pequeno para a câmara. Primeiramente efetuou-se o modelo em *SolidWorks* e chegou-se ao resultado observável na Figura 4.9.

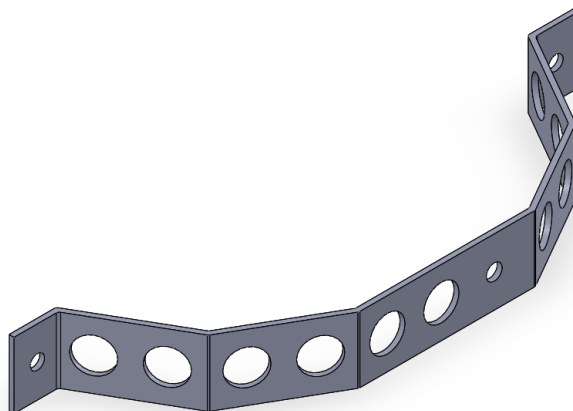


Figura 4.9 - Modelo *SolidWorks* do suporte

O local aconselhado para a colocação dos sensores de proximidade seria à altura das canelas, permitindo detetar uma pessoa deitada ou em pé. No entanto não existe nenhuma estrutura que possibilite a colocação a essa altura e não detete as pernas ou impeça a subida e descida do protótipo, juntando-se ainda a agravante de que a câmara deverá ser colocada numa posição mais elevada que permita melhor linha de visão. Tendo isto em consideração, bem como o facto de o veículo se mover a velocidades baixas e o primeiro local de eventual embate serem as rodas que absorvem grande parte do choque, tomou-se a decisão de colocar o suporte no perfil de alumínio ao nível do tampo da mesa. Esta posição permite um funcionamento correto dos sensores e da câmara uma vez que garante uma boa linha de visão e que as pernas do AGV não são detetadas pelos sensores, criando falsos positivos.

Para produzir esta estrutura foi utilizada uma chapa metálica com a altura pretendida e foi cortada, furada e quinada de acordo com o desenho apresentado no Anexo A.

Após instalação do suporte, visível na Figura 4.10, foram efetuados testes e evidenciou-se que ao ligar o motor do veículo os valores dos sensores ultrassónicos eram influenciados pela vibração do mesmo. Isto devia-se a um problema com flutuações do nível de tensão na linha de zero dos sensores, tendo sido solucionado ligando-o à estrutura da mesa.

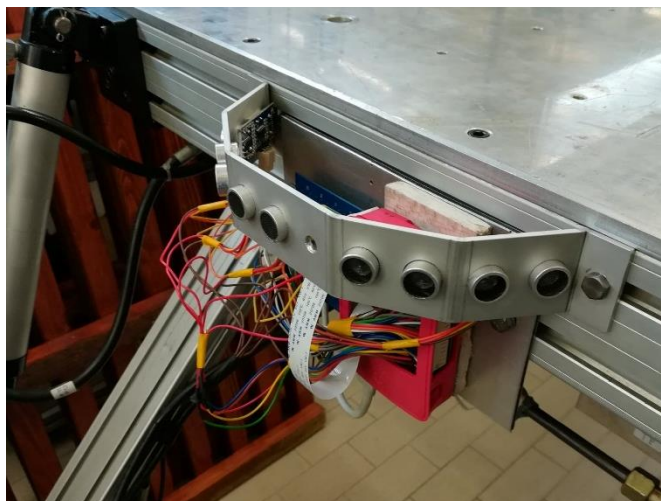


Figura 4.10 – Fixação inicial do suporte



Embora a entrada dos componentes no suporte seja efetuada com um ligeiro aperto e não exista um deslocamento causado pelo movimento da Mesa Hidráulica Móvel, achou-se por bem colocar um pedaço de fita cola de dupla face entre os componentes e o suporte para garantir a sua fixação e eliminar o contacto entre superfícies metálicas.

#### 4.4 Sistema elétrico e *hardware*

As alterações mais importantes são relativas à troca do microautómato utilizado, por um tecnologicamente mais atualizado e com melhor capacidade de comunicação, e à instalação de baterias novas. Em paralelo com estas mudanças foi também alterado o variador de frequência, por um de menor volumetria, para libertar espaço debaixo da mesa para os novos componentes, remover a parte elétrica referente aos sensores óticos e outras alterações para permitir a alimentação e comunicação com os novos componentes.

##### 4.4.1 Autómato

O *Twido* é um microautómato que se encontra desatualizado e, para permitir a evolução do protótipo, este foi substituído por um mais recente. Embora existam diversos autómatos de múltiplas marcas a escolha foi bastante simples uma vez que havia um autómato recente da marca do anterior disponível para utilização no laboratório. Este PLC é o TM251MESC da *Schneider Electric* e o seu módulo base tem duas entradas Ethernet, uma entrada *serial* (RS232/RS485), uma entrada por mini B USB e uma entrada SUB-D 9. Também tem um conjunto de LEDs que indicam o seu estado de funcionamento. Este módulo base possui um tempo de ciclo de execução de 0,3  $\mu$ s, o que é três vezes menor do que o módulo *Twido* anteriormente presente no protótipo. Porém a grande diferença a nível de tempo de processamento encontra-se na leitura das entradas analógicas, onde nos módulos *Twido* cada leitura demora 160 ms enquanto que nos módulos analógicos atuais cada leitura tem tempo da ordem de 1 ms. Uma vez que o veículo apresenta dezasseis entradas analógicas para ler esta diferença é muitíssimo significativa. É possível visualizar o módulo base TM251MESC na Figura 4.11. <sup>[11][12]</sup>



Figura 4.11 - Módulo base TM251MESC <sup>[9]</sup>

Juntamente com este componente, encontravam-se disponíveis no laboratório dois módulos úteis para o nosso projeto. Estes são o módulo de *Ethernet switch* TM4ES4, que vai permitir a comunicação entre o PLC e as *Raspberry Pi*, e o módulo de entradas digitais TM3DM24R.

Todos os componentes acrescentados ao protótipo comunicam com o PLC por cabo RJ45 por *Ethernet*, logo as entradas e saídas (analógicas e digitais) serão as mesmas com a exceção das ligações aos sensores óticos que foram retiradas. Logo foram necessários novos módulos para suportar essas conexões.

Todos os cinco módulos de expansão diferentes utilizados para este projeto estão dispostos na Figura 4.12.



Figura 4.12 - Módulos de expansão <sup>[12][13][14][15][16]</sup>

De seguida são descritos todos os módulos que compõem o PLC instalado no veículo. Estes são:

- *Módulo base TM251MESC*, corre o programa de controlo do AGV;
- *Módulo expansão TM4ES4*, módulo Ethernet switch que estabelece comunicação entre PLC e *Raspberry Pi* e *Access Point* (ver o subcapítulo 5.1);
- *Módulo expansão TM3DM24R*, com dezasseis entradas e oito saídas digitais por relé;
- *Dois módulos expansão TM3DQ16RG*, com dezasseis saídas digitais por relé cada;
- *Módulo expansão TM3AM6G*, com quatro entradas e duas saídas analógicas;
- *Dois módulos expansão TM3AI8G*, com oito entradas analógicas cada.

De modo a alterar o mínimo número de cabos no quadro elétrico, tentou-se que as funções agrupadas nos módulos anteriores se mantivessem inalteradas nestes módulos. As ligações específicas de cada módulo podem ser vistas no Anexo B, mas apresentam-se na Tabela 3 as funções de cada um dos módulos.

Tabela 3 - Funções dos diversos módulos do autómato

Módulo	Função
TM251MESC	<ul style="list-style-type: none"> <li>• Processamento do programa</li> </ul>
TM4ES4	<ul style="list-style-type: none"> <li>• Comunicação com <i>Raspberry Pi</i></li> <li>• Comunicação com <i>Access Point</i></li> </ul>
TM3DM24R	<ul style="list-style-type: none"> <li>• Monitorização da emergência</li> <li>• Atuação solenoides da direção</li> <li>• Atuação solenoides da contrapressão</li> <li>• Atuação <i>start/stop</i> do variador de frequência</li> </ul>
TM3AM6G	<ul style="list-style-type: none"> <li>• Sinal de referência para o variador de frequência</li> <li>• Leitura do inclinómetro</li> </ul>
TM3AI8G	<ul style="list-style-type: none"> <li>• Leitura dos transdutores de posição</li> </ul>
TM3AI8G	<ul style="list-style-type: none"> <li>• Leitura da carga das baterias</li> <li>• Leitura do transdutor da direção</li> <li>• Leitura dos transdutores de pressão</li> </ul>
TM3DQ16RG	<ul style="list-style-type: none"> <li>• Atuação solenoides da perna frente direita</li> <li>• Atuação solenoides da perna frente esquerda</li> </ul>
TM3DQ16RG	<ul style="list-style-type: none"> <li>• Atuação solenoides da perna trás direita</li> <li>• Atuação solenoides da perna trás esquerda</li> </ul>

#### 4.4.2 Baterias

O protótipo, aquando do início deste projeto, tinha oito baterias de 12 V que tinham uma capacidade entre 7,2 e 15 Ah, como anteriormente exposto na Figura 2.18. No entanto, era reconhecida na versão anterior uma autonomia abaixo da pretendida.

Dado o aumento do espaço disponível para as baterias e sendo requerida a sua substituição, optou-se por aumentar a autonomia ligeiramente em todos os subsistemas colocando baterias de 12V 12Ah. A única exceção encontra-se na alimentação do sub-sistema 12 V potência onde foi colocado um conjunto de baterias com o maior valor de Ampere-hora possível, dentro do orçamento e do espaço disponível.

Após estudo do espaço livre, que teve um aumento considerável devido à troca de variador de frequência (ver subcapítulo 4.4.3), e das opções existentes no mercado, selecionaram-se duas baterias do modelo LC R1233P da *Panasonic* com 12 V e 33 Ah, aumentando assim a autonomia em 36 Ah, ou seja, um aumento superior a 100%.<sup>[17]</sup>

Após esta alteração o esquema elétrico de alimentação, que é parte do esquema elétrico completo apresentado no Anexo C, apresenta a estrutura exposta na Figura 4.13.

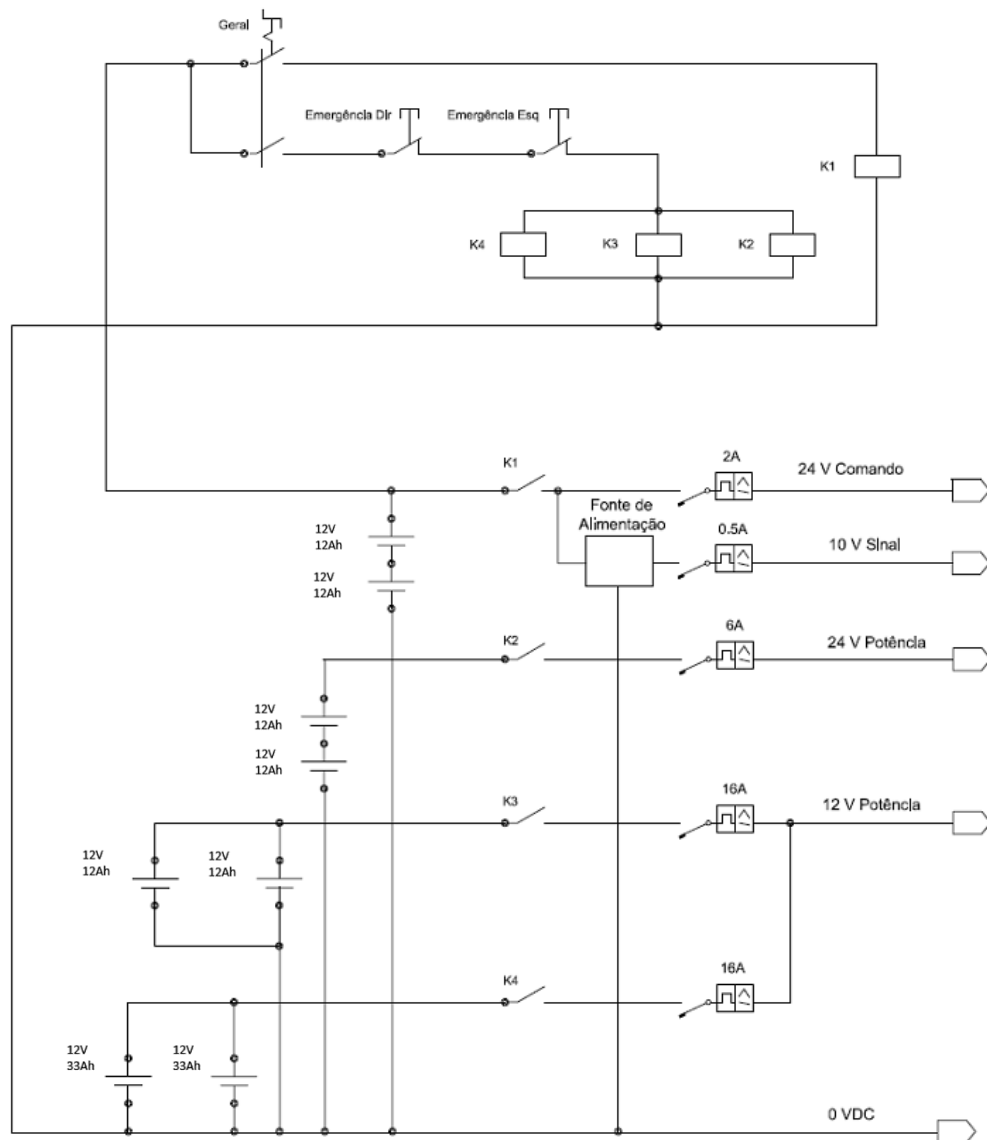


Figura 4.13 - Esquema elétrico de alimentação atual

#### 4.4.3 Variador de Frequência

O variador de frequência instalado, embora funcional, apresentava dimensões exageradas (140 x 200 x 170) em relação a outros modelos mais recentes e disponíveis.

Visto que existia disponível no laboratório um variador de frequência da marca *OMRON* modelo 3G3JV-AB004 com dimensões de 68 x 128 x 112 com características de potência iguais ao anterior, utilizou-se este libertando bastante espaço na parte inferior da mesa sem qualquer custo monetário ou a nível de desempenho.

Encontra-se na Figura 4.14 uma imagem do variador de frequência instalado na Mesa Hidráulica Móvel.4760000 974848



Figura 4.14 - Variador de frequência 3G3JV-AB004 <sup>[18]</sup>

Juntamente com a instalação física é também essencial que se configurem os parâmetros do variador para permitir um funcionamento correto. Na Tabela 4 estão apresentados os parâmetros alterados e os seus novos valores. <sup>[18]</sup>

Tabela 4 - Parâmetros configurados no variador de frequência

Parâmetro	Valor	Significado
Lo/Re	Re	<i>Remote</i>
n02	1	Entradas <i>multi-function</i> ativadas
n03	2	Referência frequência 0 - 10 V
n05	1	<i>Reverse</i> desativado
n09	50	Frequência máxima 50 Hz
n10	230	Tensão máxima 230V
n11	50	Frequência a tensão máxima 50 Hz
n32	2,0	Corrente máxima 2,0 A

#### 4.4.4 Access Point

De modo a permitir uma comunicação sem fios entre o protótipo e o computador, responsável por monitorizar e comandar o mesmo através da HMI, instalou-se um AP (*Access Point*) para permitir comunicação com o PLC por *Wi-Fi*.

Este componente tem a capacidade de transformar um sinal transmitido por *Ethernet*, pelo autómato, num sinal *Wi-Fi*, captável pelo computador e vice-versa.

Na escolha deste componente apenas se teve em consideração o custo e o tamanho, já que a velocidade e banda da comunicação requeridas pelo objeto de estudo são baixas e cumpridas por qualquer AP.

No seguimento de algum estudo sobre as opções, escolheu-se o modelo *EAP115Wall* da *TP-Link*, que se vê na Figura 4.15. <sup>[19]</sup>



Figura 4.15 - Access Point *EAP115Wall* <sup>[19]</sup>

A maioria dos AP atuais são alimentados por PoE (*Power over Ethernet*), uma tecnologia que permite a transferência de dados e alimentação através do mesmo cabo. No entanto, o módulo *switch* utilizado não possui essa facilidade o que cria a necessidade de utilizar um injetor de PoE.

#### 4.4.5 Injetor de PoE

Os injetores de PoE podem ser de várias tensões, mas a tensão *standard* é de 48 V. Foi selecionado um injetor PoE da mesma marca que o AP, *TP-Link*, e com a tensão *standard*. Este injetor escolhido foi o TL-PoE150S e pode ser visto na Figura 4.16. <sup>[20]</sup>



Figura 4.16 - Injetor PoE TL-PoE150S <sup>[20]</sup>

## 5 Aplicação informática desenvolvida

Neste capítulo proceder-se-á à exposição do desenvolvimento de todas as etapas de programação dos dispositivos que são programáveis.

Como mencionado anteriormente, a instalação de um PLC novo, com um *software* de programação diferente, e a incorporação de um novo modo de navegação tornou necessário o desenvolvimento de um novo programa de comando que leia os valores dos sensores e execute os comandos dos movimentos do AGV em função dos mesmos.

Utilizando o mesmo *software* de desenvolvimento foi ainda produzida uma interface que permitisse a leitura dos valores dos sensores e comando do veículo, quando em modo manual, bem como a seleção do modo de funcionamento do mesmo.

Foi também preciso criar de raiz um programa que fizesse o processamento de imagem e fornecesse a posição de um colete refletor, quando encontrado, e que ativasse e interpretasse as respostas dos sensores ultrassónicos.

Foi ainda necessário configurar a comunicação entre todos os componentes, quer a comunicação por cabo físico entre as *Raspberry Pi* e o PLC e entre o PLC e o AP, quer a ligação *Wi-Fi* entre o *Access Point* e o computador de supervisão e comando.

Para facilitar o seguimento do processo, este capítulo foi dividido nas seguintes secções:

- *Comunicação*, descrição da transferência de informação entre todos os componentes;
- *Processamento de imagem e sensores ultrassónicos*, programa desenvolvido para processar as imagens e ler os sensores de proximidade;
- *Programa de controlo*, explicação da lógica seguida no comando dos movimentos da Mesa Hidráulica Móvel;
- *Interface*, exposição da interface de supervisão e comando criada e seu modo de funcionamento.

### 5.1 Comunicação

Uma vez que na arquitetura de comando do veículo são considerados diferentes dispositivos programáveis com funcionamento autónomo, a partilha de informação entre estes componentes é essencial para o seu correto funcionamento. A informação processada pelos microcomputadores tem de chegar ao PLC e este tem de comunicar com o computador onde está a HMI. A arquitetura utilizada na comunicação pode ser observada na Figura 5.1.

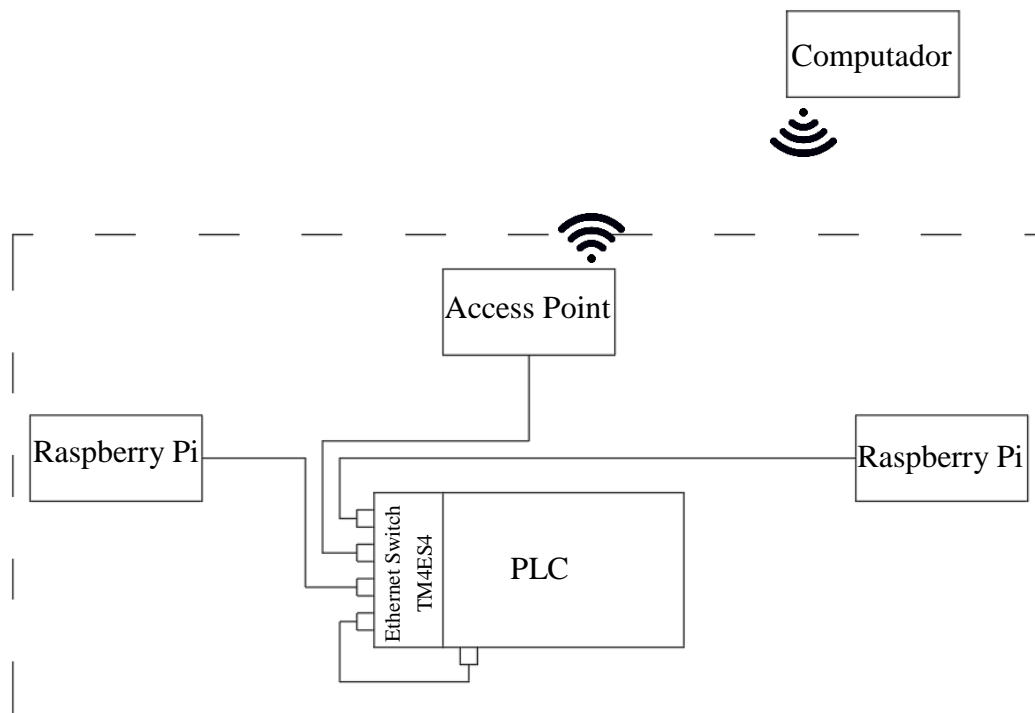


Figura 5.1 - Arquitetura de comunicação

Uma vez que foi utilizada uma comunicação por *Ethernet* foi necessário alocar IPs fixos pertencentes à mesma rede em todos os componentes envolvidos. Isto é conseguido mantendo os três primeiros valores idênticos (192.168.1) e escolhendo no último valor um número diferente para cada componente.

Toda a programação do autômato foi realizada utilizando o programa de desenvolvimento *SoMachine* (V4.1), uma versão da *Schneider Electric* de suporte à programação *Codesys*, e a definição do IP do PLC é feita no mesmo. Acedendo à janela “Ethernet\_1” é possível configurar o IP fixo, sendo visíveis na Figura 5.2 os valores colocados.

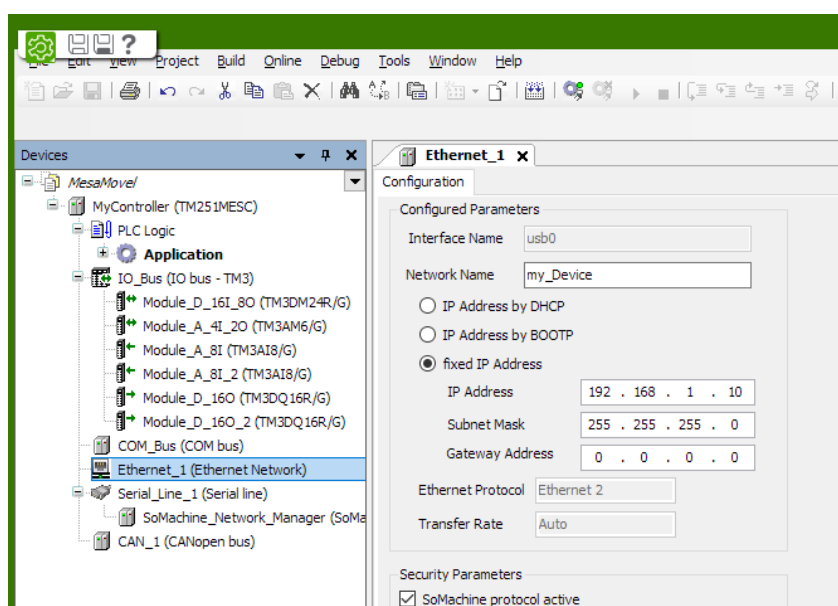


Figura 5.2 - Configuração do IP fixo no PLC

Para fazer semelhante configuração nas *Raspberry Pi* o processo é relativamente simples. Basta aceder ao ficheiro correspondente à definição dos IPs escrevendo na linha de comandos o seguinte e carregar *Enter*:

```
sudo nano /etc/dhcpd.conf
```

Isto fará aparecer um ficheiro onde se podem alterar as linhas correspondentes à parte da *Ethernet*, identificadas por *eth0*, para os valores pretendidos (192.168.1.11 para a *Raspberry Pi* da frente e 192.168.1.12 para a de trás), tal como se pode ver na Figura 5.3.

```
GNU nano 2.7.4 File: /etc/dhcpd.conf

# Respect the network MTU. This is applied to DHCP routes.
option interface_mtu

# A ServerID is required by RFC2131.
require dhcp_server_identifier

# Generate Stable Private IPv6 Addresses instead of hardware based ones
slaac private

# Example static IP configuration:
interface eth0
static ip_address=192.168.1.12/24
#static ip6_address=fd51:42f8:caae:d92e::ff/64
static routers=192.168.1.1
static domain_name_servers=192.168.1.1 8.8.8.8 fd51:42f8:caae:d92e::1

# It is possible to fall back to a static IP if DHCP fails:
# define static profile
#profile static_eth0

^G Get Help ^O Write Out ^W Where Is ^K Cut Text ^J Justify ^C Cur Pos
^X Exit ^R Read File ^M Replace ^U Uncut Text ^T To Spell ^_ Go To Line
```

Figura 5.3 - Configuração do IP fixo na *Raspberry Pi*

Uma vez introduzidos os valores pretendidos gravam-se as alterações e sai-se premindo em *Ctrl+X* seguido de *Y* (para seleccionar a opção de gravar antes de sair) seguido de *Enter*.

O último componente que requer a fixação do IP é o computador e pode ser feito indo ao Centro de Rede e Partilha, seleccionando a ligação utilizada, abrir as propriedades, escolher o “Protocolo IP Versão 4 (TCP/IPv4)”, clicar em “Utilizar o seguinte endereço IP:”, colocar os valores pretendidos, como se vê na Figura 5.4, e finalmente marcar a opção “Validar definições ao sair” e clicar em *OK*.

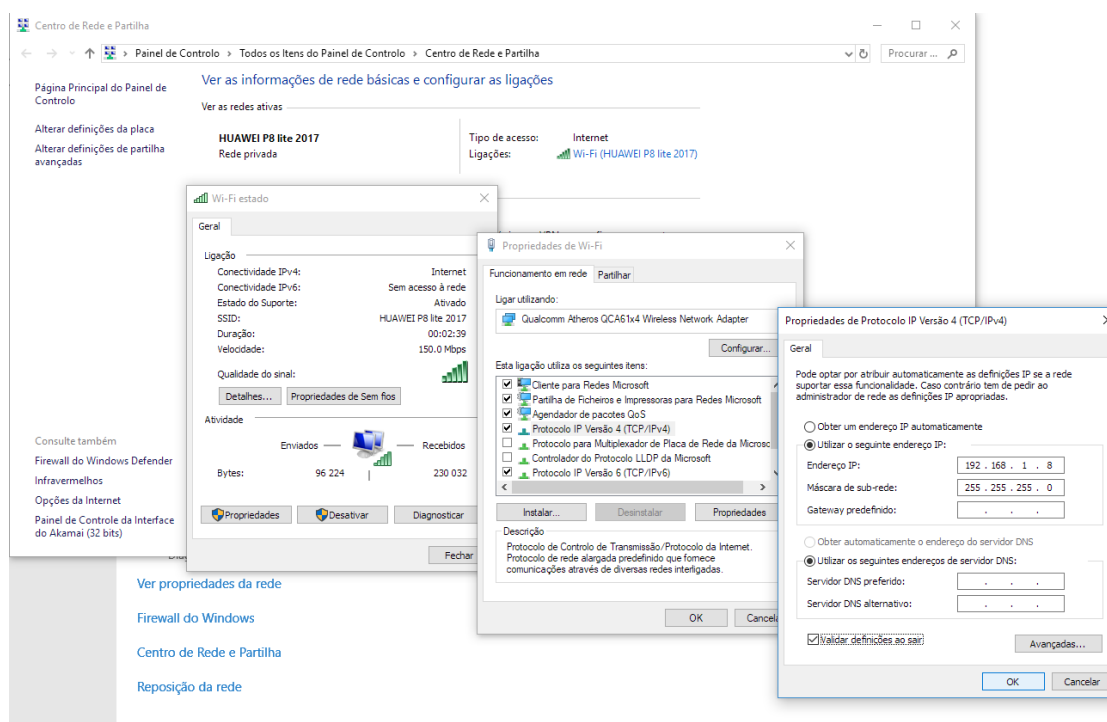


Figura 5.4 - Configuração do IP fixo no computador

Após o primeiro passo de fixar o IP de cada componente e ligar os componentes todos (o PLC, as duas *Raspberry Pi* e o *Access Point*) ao *switch* com cabos RJ45 pode-se prosseguir para a configuração do *Access Point*.

O primeiro passo na configuração do AP é a instalação do programa *Omada Controller* da *TP-Link*. De seguida é necessário colocar o IP fixo do computador em 192.168.0.100 (não esquecer de no final voltar a colocar no definido anteriormente) e iniciar sessão com o *username* e *password* que vêm de fábrica (Figura 5.5).

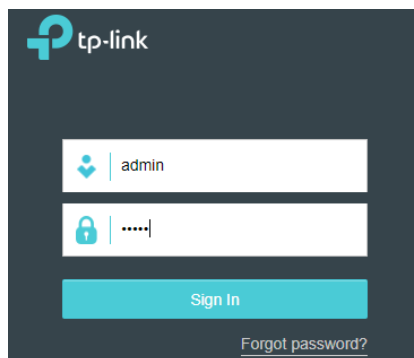


Figura 5.5 - Login *Omada Controller*

Deverá aparecer uma janela com todos os dispositivos ligados da *TP-Link* que, no presente caso, deve ser apenas o AP, no qual se deve seleccionar o componente que se pretende configurar. Uma vez escolhido o *EAP115Wall* tem que se definir qual o nome e *password* de segurança que se quer dar à rede Wi-Fi, tal como representado na Figura 5.6.

Figura 5.6 - Definição do nome e *password* da rede Wi-Fi

Por fim, cria-se uma conta de administrador para utilizar da próxima vez que se entrar no programa, na qual se devem fornecer os dados visíveis na Figura 5.7.

Figura 5.7 - Criação de conta no *Omada Controller*



Estando o *Access Point* configurado e o *software Omada Controller* aberto é apenas necessário aceder à *Wi-Fi* com o nome e *password* anteriormente escolhidos.

A comunicação entre as *Raspberry Pi* e o PLC é efetuada utilizando *Modbus TCP/IP*, onde o microcomputador vai escrever variáveis inteiras nos *holding registers* do autómato que apenas tem de aceder aos valores escritos nos endereços corretos. As partes desta comunicação relativas a cada interveniente serão expostas com maior detalhe nos subcapítulos seguintes.

## 5.2 Processamento de imagem e sensores ultrassónicos

O primeiro passo para poder fazer qualquer tipo de operação no microcomputador é a instalação de um sistema operativo sobre o qual trabalhar. Embora existam sistemas operativos para a *Raspberry Pi* de outros fornecedores, o oficial é o *Raspbian* que vai na versão *Stretch* 4.14 e foi este o instalado e tem a interface apresentada na Figura 5.8.

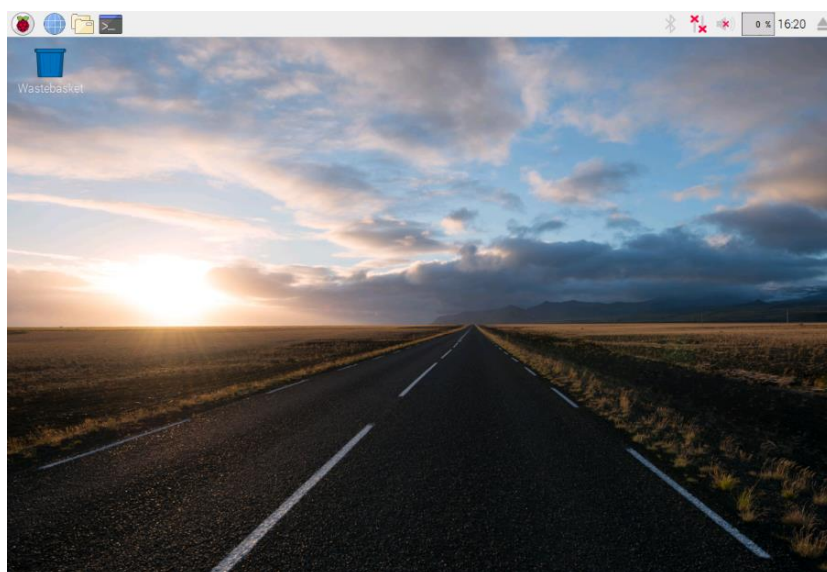


Figura 5.8 - Ambiente de trabalho do *Raspbian* versão *Stretch* 4

Para além do sistema operativo, foi também requerida a instalação de algumas bibliotecas que possuem funções que são utilizadas no programa de processamento e comunicação.

O programa foi redigido na linguagem *Python*, pois esta é a linguagem padrão para este tipo de microcomputadores. Esta linguagem de programação tem grande potencial, não sendo difícil de aprender, tendo um nível de programação similar ao *Javascript*.

Como se pretende que o programa funcione continuamente criou-se no mesmo um *loop* infinito. Assim sendo o programa está estruturado da seguinte forma:

- Chamada das bibliotecas;
- Inicialização de variáveis;
- *Loop* infinito.

Embora o programa corrido se encontre em apenas um ficheiro e não seccionado, para mais fácil compreensão da lógica de programação vai ser dividido este subcapítulo em três secções:

- Visão artificial;
- Sensores de proximidade;
- Comunicação e segurança.

### 5.2.1 Visão artificial

A decisão tomada sobre como fazer a navegação recaiu, como dito anteriormente, sobre o seguimento de um alvo. Este alvo será um colete refletor de segurança.

Devido à sua cor distinta das presentes num ambiente comum e, particularmente, do laboratório onde o AGV irá circular, optou-se por fazer a deteção do colete baseada em cor.

As bibliotecas que tiveram de ser instaladas e importadas para esta parte do programa foram:

- *imutils*, que permite criar uma leitura continua da imagem da câmara sem necessidade de gravar em disco;
- *OpenCV*, uma biblioteca muitíssimo usada no tratamento de imagem e que permite efetuar múltiplas funções;
- *numpy*, que permite o tratamento de imagens como um *array*;
- E é necessário importar o tempo (*time*).

O que é necessário inicializar para esta parte do programa são os parâmetros segundo os quais a imagem é capturada e encontram-se dispostos no código abaixo.

```
> usingPiCamera = True
> frameSize = (800, 608)
> vs = VideoStream(src=0, usePiCamera=usingPiCamera,
    resolution=frameSize, framerate=32).start()
> time.sleep(2.0)
```

As duas primeiras linhas (marcadas por “>”) servem para inicializar os valores que vão ser usados na terceira linha e definem que se vai utilizar a *Raspberry Pi Camera*, em detrimento de uma *webcam*, e que o tamanho da imagem será 800 por 608 pixel, tendo sido tomada esta relação de valores para não criar uma distorção na imagem, e destes valores em particular para permitir a visualização de três janelas (*frame*, *mask* e *median*) no ecrã aquando dos testes de calibração do programa. A terceira linha serve para iniciar a transferência de imagens contínua de acordo com os parâmetros anteriormente escolhidos. Já a última linha de código gera um período de espera de dois segundos para deixar a câmara aquecer.

O código relativo ao processamento da imagem que se encontra dentro do *loop* principal é o seguinte.

```
> frame = vs.read()
> lower = np.array([0,145,130])
> upper = np.array([85,255,235])
> mask = cv2.inRange(frame, lower, upper)
> median = cv2.medianBlur(mask,15)
> contours = cv2.findContours(median, 1, 2)
> cnt = contours[0]
> x,y,w,h = cv2.boundingRect(cnt)
```

O primeiro comando retira um *frame* do *stream* de imagens para ser analisado. A segunda e terceira linhas definem o que são os limites inferior e superior, respetivamente, do espectro de cores a ser detetado. Note-se que os valores das cores são dados em código BGR (*Blue Green Red*) e que estes são os parâmetros mais importantes, uma vez que têm de se definir limites que abranjam as várias tonalidades da cor do colete, causadas pelas sombras, sem detetar outras cores similares, como o amarelo e o verde. A quarta linha programada utiliza a função *inRange* da biblioteca *OpenCV* que cria uma imagem (*mask*) a branco e preto (uns e zeros respetivamente), onde os pixel com uma cor dentro dos limites é colocado a branco e os restantes a preto. A linha número cinco usa a função *medianBlur* que homogeneíza a imagem *mask* e elimina possíveis falsos positivos causados pela iluminação. A função *findContours* identifica os contornos da imagem *median*. A sétima linha seleciona a informação necessária do vetor devolvido pelo comando anterior. Por fim a função *boundingRect* encontra um retângulo que rodeie os contornos e devolve o ponto de início desse retângulo ( $x$  e  $y$ ), a sua largura ( $w$ ) e altura ( $h$ ).

Embora apenas seja necessário saber os parâmetros do retângulo que contem o colete refletor para o programa, durante os testes foi necessário visualizar todas as etapas tendo sido acrescentadas algumas linhas de código para essa fase que desenhavam o retângulo na imagem *frame* e que mostravam no ecrã as imagens *frame*, *mask* e *median*. O resultado deste código de teste encontra-se exemplificado na Figura 5.9.

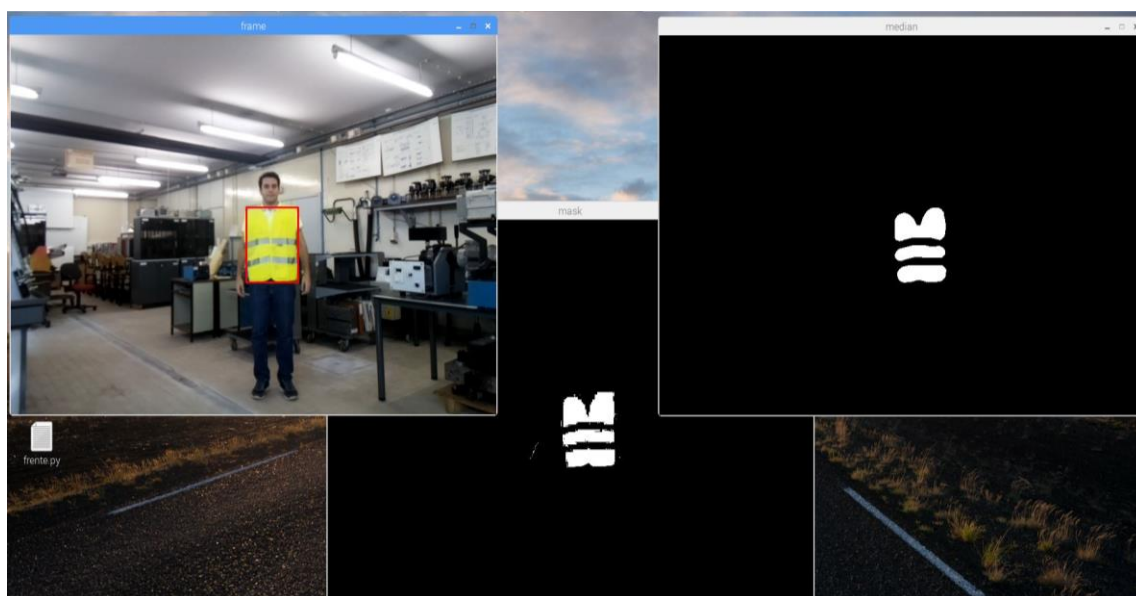


Figura 5.9 - Teste de deteção do colete refletor

Com este código determina-se a posição do colete e esta informação pode ser enviada para o PLC.

### 5.2.2 Sensores de proximidade

O funcionamento destes sensores já foi explicado no subcapítulo 4.2 e o seu programa consiste apenas na tradução do explicado em código.

Para esta parte do código é necessário importar o tempo e a biblioteca *RPi.GPIO*, que contém as funções relativas à utilização dos pinos GPIO.

Em primeiro lugar é necessário inicializar cada pino GPIO como se pode ver no código abaixo, sendo necessário escolher qual o pino que corresponde a cada variável, e definir se é uma entrada (*IN*) ou saída (*OUT*) e, no caso de ser saída, qual o seu valor.

```

> GPIO.setmode(GPIO.BCM)
> T1T=05
> T1E=23
> T2T=06
> T2E=24
> T3T=13
> T3E=25
> T4T=19
> T4E=12
> T5T=26
> T5E=16
> GPIO.setup(T1T,GPIO.OUT)
> GPIO.setup(T1E,GPIO.IN)
> GPIO.setup(T2T,GPIO.OUT)
> GPIO.setup(T2E,GPIO.IN)
> GPIO.setup(T3T,GPIO.OUT)
> GPIO.setup(T3E,GPIO.IN)
> GPIO.setup(T4T,GPIO.OUT)
> GPIO.setup(T4E,GPIO.IN)
> GPIO.setup(T5T,GPIO.OUT)
> GPIO.setup(T5E,GPIO.IN)
> GPIO.output(T1T, False)
> GPIO.output(T2T, False)
> GPIO.output(T3T, False)
> GPIO.output(T4T, False)
> GPIO.output(T5T, False)

```

Com as variáveis definidas e associadas aos GPIO é possível avançar para o programa referente aos sensores que se encontra dentro do *loop*. Este programa é composto por cinco blocos idênticos, um para cada sensor, e um desses blocos encontra-se exposto abaixo.

```

> GPIO.output(T1T, True)
> time.sleep(0.00001)
> GPIO.output(T1T, False)
> while GPIO.input(T1E)==0:
>     pulse_start_T1 = time.time()
> while GPIO.input(T1E)==1:
>     pulse_end_T1 = time.time()
> pulse_duration_T1=pulse_end_T1 - pulse_start_T1
> distance_T1 = pulse_duration_T1 x 171500
> distance_T1 = int(round(distance_T1,0))

```

As primeiras três linhas correspondem ao envio de um impulso de 10  $\mu$ s, onde a primeira linha ativa o sinal, a segunda espera o tempo pretendido e a terceira retorna o sinal a zero. O *loop* da quarta e quinta linhas vai guardando o tempo atual até o sinal da linha de *echo* comutar para “1”. O *loop* da sexta e sétima linha tem o mesmo objetivo do anterior, mas quando o sinal descer a “0”. O código da linha oito calcula o tempo em *high* da linha de *echo*. A nona linha faz o cálculo da distância pela fórmula anteriormente explicada. Por fim a distância calculada (em milímetros) é arredondada à casa das unidades e convertida num inteiro para permitir a sua comunicação.

### 5.2.3 Comunicação e segurança

A última componente necessária para o funcionamento do programa é a comunicação das variáveis com o PLC. Embora não seja essencial para os programas, é extremamente recomendada a inserção de uma medida de segurança que permita detetar uma falha na comunicação, tendo esta de ser algo implementado entre os dois sistemas envolvidos na comunicação.

A solução encontrada para este sistema de segurança, vulgarmente denominada *watchdog*, de modo a não implicar outra função de comunicação (usar apenas inteiros) foi a de alterar uma variável a todos os ciclos no programa do microcomputador e, na eventualidade da existência de uma quebra na comunicação, o PLC detetá-la-ia pela permanência da variável de segurança num dos valores. Para tal foi necessário inicializar a variável de segurança, denominada *dog*, correndo a seguinte linha de código.

```
> dog = 0
```

Durante o *loop* infinito foi necessário alterar o seu valor a cada ciclo, tendo-se recorrido ao seguinte código para esse efeito.

```
> if dog == 0:
>     dog = 5
> elif dog == 5:
>     dog = 0
```

Isto permite uma alternância contínua do valor de *dog* entre 0 e 5 que, associada ao programa do PLC, possibilita a deteção de falhas na comunicação.

A comunicação, como referido anteriormente, será feita por *Modbus TCP/IP* e a maioria do trabalho será efetuado de um modo transparente pelo microcomputador, tendo o PLC apenas de ler as variáveis do endereço.

No programa da *Raspberry Pi* recorre-se à biblioteca *pyModbusTCP*, mais propriamente à parte *ModbusClient*, que tem funções que permitem ler e escrever valores num outro endereço IP da mesma rede.

Para tal têm que se definir alguns parâmetros na parte de inicial do programa, podendo-se analisar esta parte do código nas linhas abaixo.

```
> SERVER_HOST = "192.168.1.10"
```

```

> SERVER_PORT = 502
> c = ModbusClient()
> c.host(SERVER_HOST)
> c.port(SERVER_PORT)

```

O código acima define a variável *c* como o cliente *Modbus* tendo como *host* o autómato e usando a porta predefinida (502).

Durante o *loop* principal o programa relativo à comunicação encontra-se no fim do mesmo e é composto pelas seguintes linhas de código.

```

> if not c.is_open():
>     if not c.open():
>         print("unable to connect to " + SERVER_HOST + ":" +
str(SERVER_PORT))
> if c.is_open():
>     comun=
[distance_T1,distance_T2,distance_T3,distance_T4,distance_T5,x,w,h,dog]
>     c.write_multiple_registers(110, comun)

```

As três primeiras linhas podem ser traduzidas em: se a ligação não estiver aberta tentar abrir; se não conseguir, emitir uma mensagem a dizer que foi impossível.

Nas outras três linhas lê-se que: se a ligação estiver aberta o vetor a comunicar é composto pelas distâncias dos cinco sensores, as dimensões e localização do retângulo e o valor do *dog* e que este vetor será escrito nos endereços 110 e seguintes. Note-se que o código da outra *Raspberry Pi* é, em tudo, idêntico ao desta à exceção do valor do endereço inicial que passa a ser 119.

Juntando todas as secções do programa é apenas necessário acrescentar uma condição de saída do *loop* que, embora não se pretenda utilizar, é necessária para o funcionamento do programa.

```

> key = cv2.waitKey(1) & 0xFF
> if key == ord("q"):
>     break

```

Este código possibilita a saída do *loop* infinito quando pressionada a tecla *Q*.

Por fim é necessário mandar executar o programa de modo automático quando o microcomputador liga. Para tal, é preciso aceder a um ficheiro que é lido no processo de *boot* do sistema e adicionar o comando de execução do programa.

Pode-se aceder ao ficheiro correto colocando na linha de comandos o seguinte.

```

> sudo nano /etc/rc.local

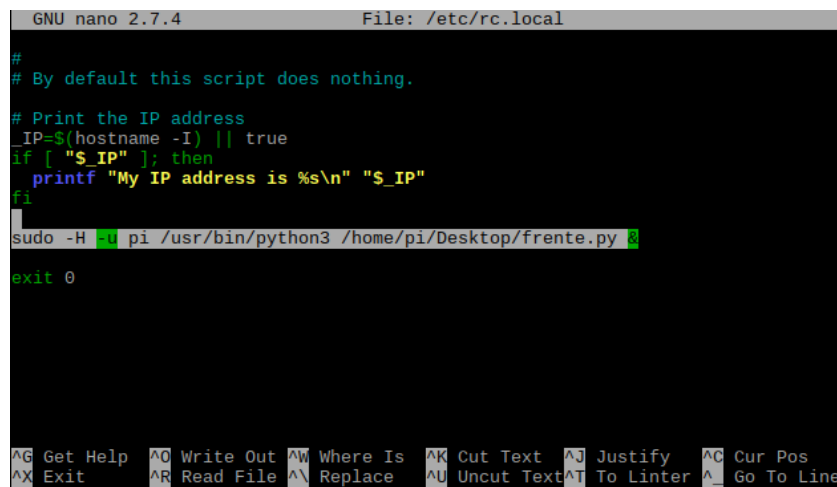
```

Isto faz abrir o ficheiro, no qual se deve descer até ao fim do mesmo e colocar o código mostrado abaixo e de seguida gravam-se as alterações e sai-se premindo em *Ctrl*+*X* seguido de *Y*, seguido de *Enter*.

```
> sudo -H -u pi /usr/bin/python3 /home/pi/Desktop/frente.py &
```

Note-se que é preciso colocar o código “*-H -u pi*” para ser possível chamar as bibliotecas, colocar a localização completa do programa a correr e colocar “*&*” no fim para o programa correr em segundo plano e não impedir a inicialização da Raspberry Pi visto que o programa é um *loop* infinito.

É possível ver o ficheiro e onde escrever a linha de código na Figura 5.10.



```
GNU nano 2.7.4 File: /etc/rc.local
#
# By default this script does nothing.
# Print the IP address
_IP=$(hostname -I) || true
if [ "$_IP" ]; then
    printf "My IP address is %s\n" "$_IP"
fi
sudo -H -u pi /usr/bin/python3 /home/pi/Desktop/frente.py &
exit 0
```

Figura 5.10 - Ficheiro lido na inicialização

Fica assim finalizado o código que correrá em cada uma das *Raspberry Pi*, que determina as distâncias medidas em cada sensor, a localização do colete e efetua a comunicação da informação com o PLC.

### 5.3 Programa de controlo dos movimentos do veículo

Este programa é residente no autómato e é o responsável pelo controlo de todos os movimentos executados pelo veículo. É através deste programa que se leem as entradas digitais e analógicas e se efetuam os comandos digitais e analógicos a executar.

Devido à complexidade, modularidade e extensão do programa, foi entendido dividi-lo em subprogramas tendo cada um uma função no programa geral. Dada a natureza do projeto achou-se por bem utilizar duas linguagens distintas nos subprogramas:

- Texto estruturado;
- SFC (*Sequential Function Chart*), semelhante ao *Grafcet*.

O primeiro passo na programação foi a identificação dos componentes usados, tal como a identificação das entradas e saídas por ele utilizadas. A lista dos módulos inseridos na programação encontra-se representada na Figura 5.11.

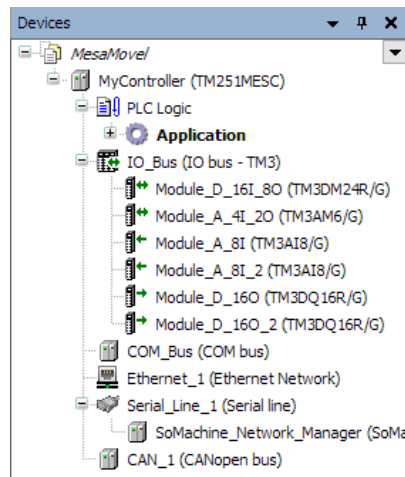


Figura 5.11 - Identificação dos módulos usados no programa

Tendo esta etapa concluída pode-se avançar para a criação dos subprogramas. Para mais fácil entendimento da lógica da programação e do funcionamento do AGV este subcapítulo foi separado em secções que correspondem a cada subprograma. É possível identificar cada subprograma e a sua ordem na Figura 5.12 seguinte.

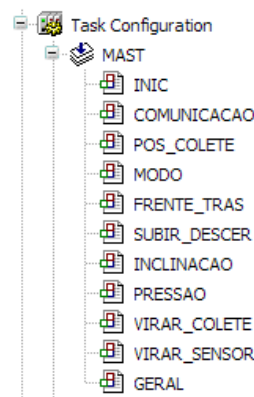


Figura 5.12 - Lista ordenada dos subprogramas

### 5.3.1 Subprograma *INIC*

Este código encontra-se escrito em texto estruturado e é responsável pela inicialização dos valores das saídas, colocando-os nas posições de segurança. Na inexistência de uma ordem mais à frente no programa garante-se um retorno à posição de segurança no início de cada ciclo.

A Mesa Hidráulica Móvel foi dotada de dois modos de funcionamento. É também neste subprograma que é definida a velocidade base para cada um deles.

A última tarefa executada por esta parte do código é a conversão do valor do transdutor de posição da direção para o ângulo de viragem da mesa, pois este é mais intuitivo para o utilizador.

### 5.3.2 Subprograma *COMUNICACAO*

Como sugerido pelo nome, o código desta parte tem como função agilizar a comunicação com as *Raspberry Pi*. Como indicado anteriormente, o PLC não tem grande tarefa na comunicação tendo apenas de associar as variáveis aos endereços corretos.

Para definir qual o endereço correspondente a cada variável utiliza-se a função *AT*, tal como se pode ver na Figura 5.13.



```

1  PROGRAM COMUNICACAO
2  VAR
3      SF1 AT %MW110: INT;
4      SF2 AT %MW111: INT;
5      SF3 AT %MW112: INT;
6      SF4 AT %MW113: INT;
7      SF5 AT %MW114: INT;
8      XF AT %MW115: INT;
9      WF AT %MW116: INT;
10     HF AT %MW117: INT;
11     DOGF AT %MW118: INT;
12     ST1 AT %MW119: INT;
13     ST2 AT %MW120: INT;
14     ST3 AT %MW121: INT;
15     ST4 AT %MW122: INT;
16     ST5 AT %MW123: INT;
17     XT AT %MW124: INT;
18     WT AT %MW125: INT;
19     HT AT %MW126: INT;
20     DOGT AT %MW127: INT;
21 END_VAR

```

Figura 5.13 – Correspondência das variáveis com os endereços de comunicação

Note-se que este subprograma foi feito em texto estruturado e que todos os valores das variáveis comunicadas são inteiros e encontram-se colocados nos endereços definidos pelos microcomputadores.

### 5.3.3 Subprograma *POS\_COLETE*

Tendo sido criado numa linguagem de texto estruturado, compete a este subprograma efetuar os cálculos para determinar, com base nas dimensões do retângulo que envolve o colete refletor, qual a distância relativa à câmara e o ângulo lateral que faz em relação à posição central.

Para determinar qual a distância a que um objeto se encontra pode -se ir pela relação entre o seu tamanho real e o tamanho aparente (Figura 5.14).

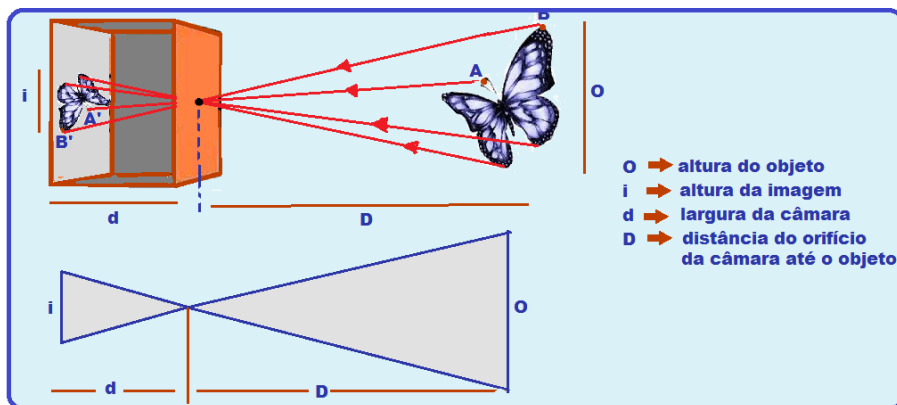


Figura 5.14 - Relação entre distâncias e alturas reais e aparentes <sup>[21]</sup>

Como é observável existe uma relação de semelhança de triângulos entre as medidas reais e as aparentes. Desta relação é possível retirar a seguinte equação.

$$\frac{D}{O} = \frac{d}{i} \Leftrightarrow$$

$$D = \frac{d \times O}{i} \Leftrightarrow$$

$$D = \frac{R}{i}$$

Embora se pudesse retirar o valor  $d$  da ficha técnica da *Raspberry Pi Camera*, medir a altura do colete e efetuar a conversão entre milímetros e pixéis (pois temos o valor de  $i$  em pixéis), foi tomada a opção de obter o valor de  $R$  por via experimental. Para tal foram retirados os valores de  $i$  para várias distâncias  $D$  e obteve-se um valor de  $R$  de 395000.

Com este cálculo determina-se o valor de  $D$  o que permite não só manter a distância pretendida do alvo, como veremos mais à frente, mas também definir qual a velocidade do motor, sendo esta tão maior quanto a diferença entre a distância pretendida e a do alvo.

Para determinar o ângulo relativo à posição central foi utilizada trigonometria simples.

Este subprograma é também responsável pela determinação da variável que identifica a presença do colete na frente da mesa ou atrás.

#### 5.3.4 Subprograma *MODO*

O veículo tem dois modos de funcionamento: o “modo manual” e o “modo autónomo”, sendo que o modo manual é ainda dividido no “modo base” e no “modo avançado”.

De forma a identificar qual o modo ativo em cada momento foi utilizado um subprograma em SFC com a lógica do esquema da Figura 5.15.

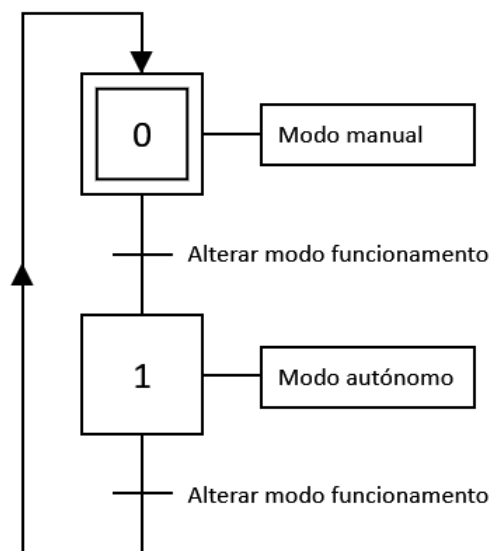


Figura 5.15 - *Grafcet* do modo de funcionamento

#### 5.3.5 Subprograma *FRENTE\_TRAS*

Este código foi escrito em SFC e é o que comanda os motores hidráulicos que realizam o movimento de translação do veículo. A lógica de programação foi feita com base no esquema da Figura 5.16.

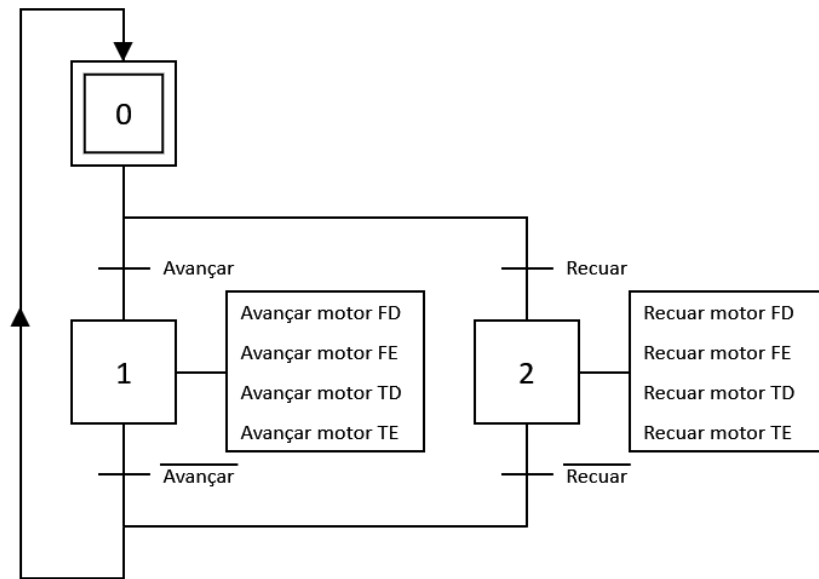


Figura 5.16 - *Grafcet* de controlo dos motores hidráulicos

O comando para avançar pode ser feito, no modo manual, por ativação de um botão da interface do operador ou, no modo autónomo, pela distância a que o veículo está do operador com colete. Estando o colete à frente a uma distância superior à desejada o veículo avança. Estando o colete atrás a uma distância inferior à desejada, também avança. O comando para recuar é executado pelas condições inversas.

Note-se que foi incluída uma característica de histerese pois, como o funcionamento é *on/off*, a sua inexistência levaria a potenciais instabilidades.

É igualmente importante referir que aquando da realização de qualquer movimento de um atuador de uma perna é também preciso atuar a válvula de carga dessa mesma perna. Se isto não for feito o circuito mantém-se em descarga e não possui capacidade de movimentação.

### 5.3.6 Subprograma *SUBIR\_DESCER*

A secção do programa aqui exposta é a que efetua a subida ou a descida do tampo da mesa através do comando dos atuadores de elevação e foi feita em SFC. Este tipo de movimento é apenas realizado no modo manual visto que não existe necessidade para o mesmo no modo autónomo. A lógica de programação encontra-se esquematizada na Figura 5.17 e é bastante similar à do subprograma anterior.

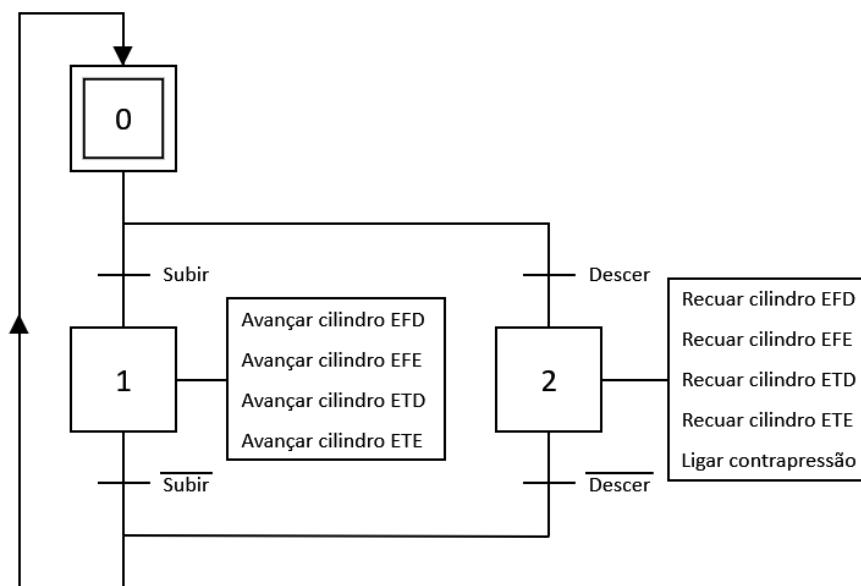


Figura 5.17 - Grafcet de controlo da elevação

Repare-se que aquando da descida do AGV é efetuada a ativação da contrapressão, permitindo uma descida mais suave.

### 5.3.7 Subprograma *INCLINACAO*

O controlo da horizontalidade do veículo foi feito em SFC e segue o esquema visível na Figura 5.18. Este controlo é principalmente usado no modo autónomo, mas foi criada uma opção no modo manual que permite nivelar o protótipo mantendo pressionado um botão. Como é observável existe uma primazia da inclinação lateral, em relação à longitudinal, visto que esta é a que maior instabilidade pode causar no protótipo.

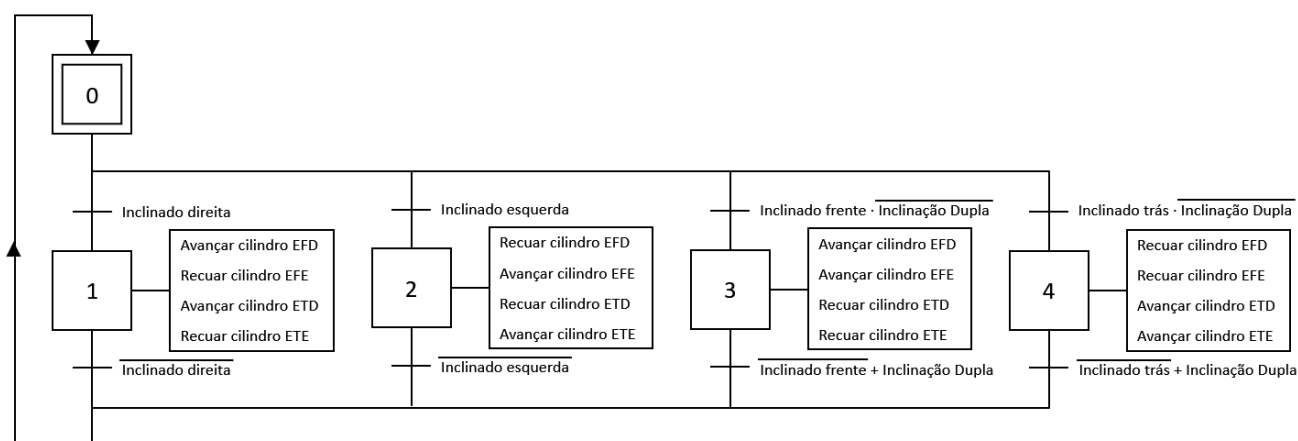


Figura 5.18 - Grafcet de controlo da inclinação

Note-se que de modo a equilibrar apenas a mesa e não proceder à sua subida ou descida, o controlo da horizontalidade é feito subindo os atuadores de elevação do lado mais baixo e descendo os do lado mais elevado. Deste modo tenta-se manter a horizontalidade da mesa, bem como a sua altura média.

### 5.3.8 Subprograma *PRESSAO*

Com o propósito de identificar e corrigir situações em que uma das pernas fique em suspenso, ou apenas encostada ao solo, durante a navegação no modo autónomo foi desenvolvido em SFC o subprograma representado esquematicamente na Figura 5.19.

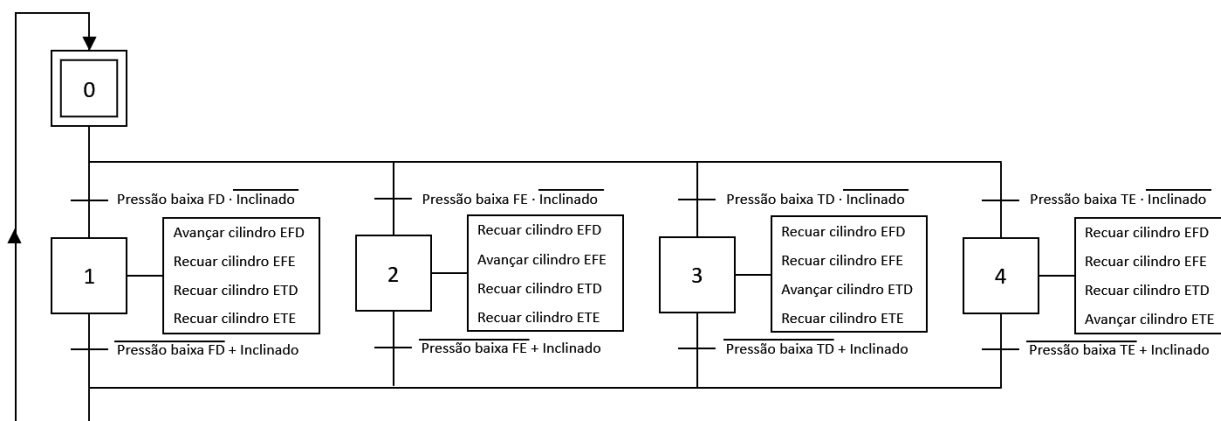


Figura 5.19 - *Grafcet* de controlo da pressão

Repare-se que qualquer correção das pressões é apenas iniciada na ausência de desvios da posição estável da inclinação, isto deve-se ao facto de a inclinação ser uma situação potencialmente perigosa que, se deixada sem supervisão, pode levar à queda do veículo.

Note-se novamente que foi efetuado um movimento de subida e descida dos cilindros de elevação numa tentativa de manter uma altura constante aquando do modo de funcionamento autónomo.

### 5.3.9 Subprograma *VIRAR\_COLETE*

Este subprograma, escrito na linguagem SFC, permite controlar a direção da Mesa Hidráulica Móvel e é fundamentado no esquema mostrado na Figura 5.20

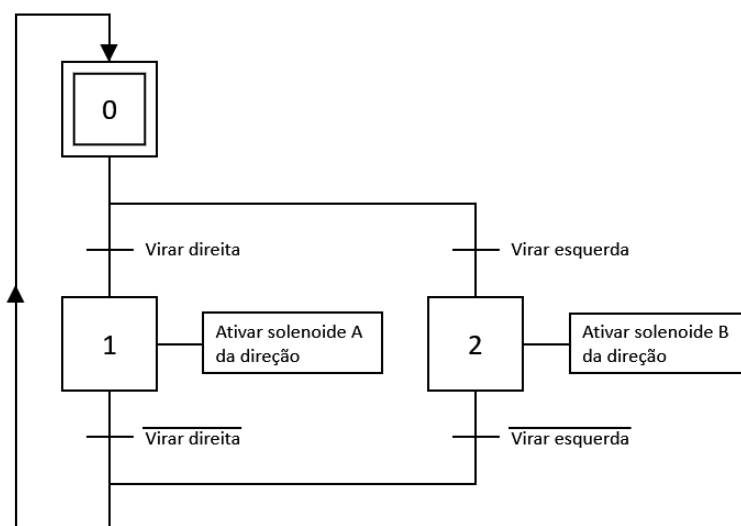


Figura 5.20 - *Grafcet* de controlo da direção

A ordem de virar pode ser dada, no modo manual base, pela ativação de um botão na consola do operador. No modo manual avançado, é possível a introdução do ângulo de viragem pretendido. No modo de funcionamento autónomo, a direção é realizada pelo desvio do colete do operador em relação à posição central

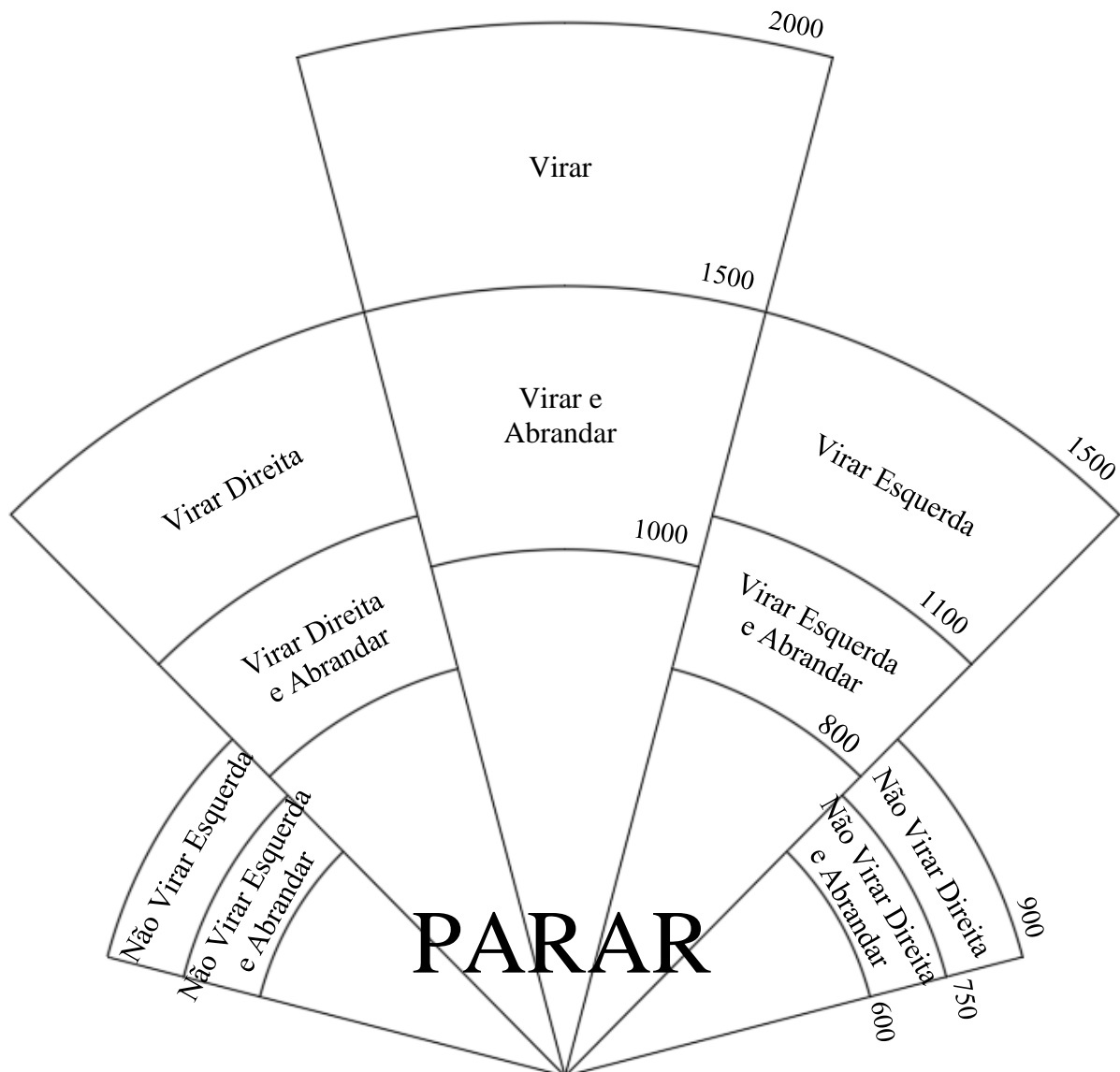
### 5.3.10 Subprograma *VIRAR\_SENSOR*

Como explicado no subcapítulo 4.2, o uso dos sensores de proximidade não será só para imobilizar o veículo, mas também para permitir um desvio de potenciais obstáculos no percurso. Para tal, foi criada uma grelha que informa o veículo de quando alterar a sua posição.

Primeiro foi colocado um filtro nos sinais provenientes das *Raspberry Pi*, relativos aos sensores ultrassónicos HC-SR04, de forma a eliminar algum do ruído e das oscilações que apresentam as suas leituras.

Com base nessas leituras é possível agir de acordo com os obstáculos encontrados, sendo importante realçar que, como sabemos a posição do operador que tem o colete vestido, é possível filtrar as deteções que ele induz nos sensores de proximidade.

Este subprograma encontra-se incumbido de fazer esse controlo e foi escrito em texto estruturado. A lógica sobre a qual foi estruturado este programa é visível na grelha apresentada na Figura 5.21, estando as medidas em milímetros.



O comando de virar dado pelo sensor central não é para um lado fixo, se existir alguma outra deteção será mandado virar para o lado concordante com a mesma, se não tiver outra deteção manda virar para a esquerda.

Ao ter a ordem de parar, o veículo é impedido de se mover naquele sentido e coloca-se o motor a baixa velocidade, permitindo apenas que o veículo se afaste lentamente do objeto.

Os comandos relativos aos sensores que se encontram na parte de trás da mesa são iguais tendo apenas os sentidos invertidos, ou seja, virar para a esquerda em vez da direita.

### 5.3.11 Subprograma *GERAL*

Este é a última parte do programa a correr, sendo que qualquer ordem aqui dada sobrepõe-se às anteriormente recebidas. É, portanto, lógico que seja aqui que se encontrem todos os comandos relativos a segurança.

Esta secção de código inicia-se pelos comandos criados pelo modo manual avançado que permitem a movimentação de cada atuador individualmente.

De seguida encontram-se as linhas que soltam as rodas quando anteriormente foi ordenado um movimento que o requeresse. Exemplo disso são os movimentos de descida e subida que são executados com as rodas frontais soltas para permitir um movimento suave e as rodas traseiras presas para impedir o movimento do veículo num plano inclinado.

Por fim encontra-se a segurança, onde se começa por identificar se não existe falha de comunicação. De seguida têm-se as últimas linhas de código que, na eventualidade de ser assinalada uma situação de emergência na botoneira física, na botoneira virtual da interface ou na comunicação aquando do funcionamento em modo autónomo, colocam todas as variáveis nos valores que correspondem ao estado de segurança da Mesa Hidráulica Móvel.

## 5.4 Interface

Para se poder comandar a Mesa Hidráulica Móvel foi necessário desenvolver uma interface para diálogo com o operador. Embora existam aplicações especializadas para a sua criação, o *SoMachine* apresenta a possibilidade de criar interfaces no mesmo programa do controlo, o que apresenta uma grande vantagem a nível de simplificação da comunicação.

Com o objetivo de tornar a interação do utilizador com a HMI o mais agradável e simples possível foi criada uma página principal que se encontra sempre presente, independentemente do modo de funcionamento.

Esta base é constituída pela leitura dos sensores de posição, de pressão, de direção e de inclinação. Pode-se ainda seleccionar o modo de funcionamento pretendido. Existe também uma botoneira que permite ligar ou desligar o motor elétrico a qualquer momento. Por último, existe um painel de monitorização do estado de carga das baterias e uma botoneira de emergência virtual.

Na Figura 5.22 é possível observar o design utilizado para a base da interface e, em conjunto com a Tabela 5, identificar a função dos diversos componentes.

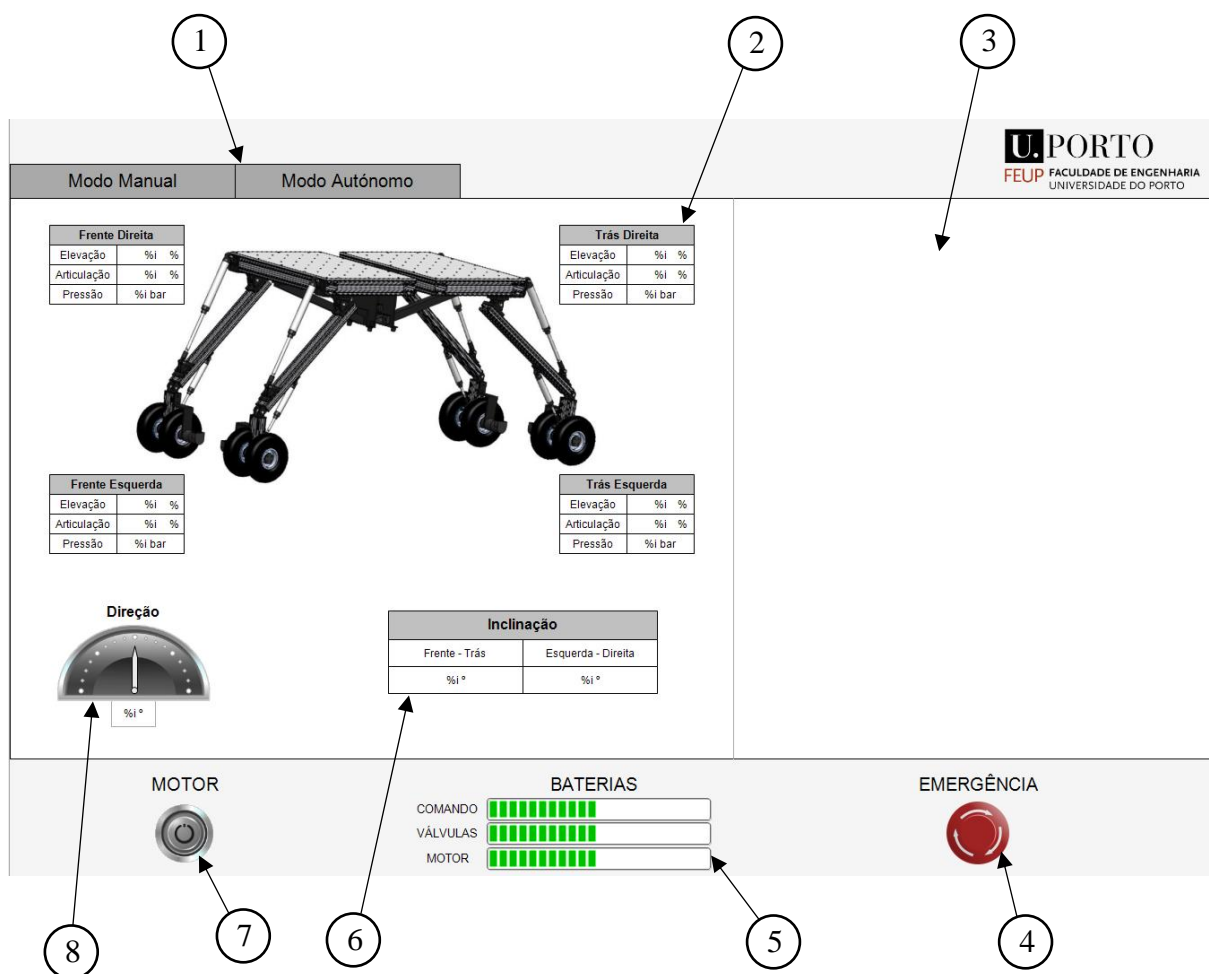


Figura 5.22 - Página base da interface

Tabela 5 - Legenda das funções da página base da interface

Número	Função
1	• Selecionar o modo de funcionamento
2	• Leitura da posição e pressão de cada perna
3	• Zona dependente do modo de funcionamento
4	• Botoneira virtual de emergência
5	• Leitura da carga das baterias
6	• Leitura da inclinação
7	• Comando do motor elétrico
8	• Leitura da direção

Note-se que é no local assinalado pelo número 3 que serão apresentadas visualizações diferentes de acordo com o modo de funcionamento selecionado no momento.

Independentemente do modo de funcionamento, quando pressionada a botoneira de emergência, real ou virtual, aparecerá uma mensagem a informar que se entrou no modo de emergência e permanecerá desse modo até a botoneira ser desatuada. Na Figura 5.23 mostra-se como fica a interface no estado de emergência.





Figura 5.23 - Interface no estado de emergência

A janela correspondente ao modo de funcionamento pode assumir três aspetos:

- Modo manual base;
- Modo manual avançado;
- Modo autónomo.

#### 5.4.1 Modo manual base

Este modo apresenta um modo de condução simples, mas que permite efetuar todos os movimentos básicos do veículo. Tem como possibilidade fazer mover o protótipo para a frente e para trás, alterar a sua direção, subir e descer o seu nível de elevação, nivelar a mesa e ainda inverter o sentido de translação. Todas estas funções se encontram visíveis na Figura 5.24 e identificadas na Tabela 6.

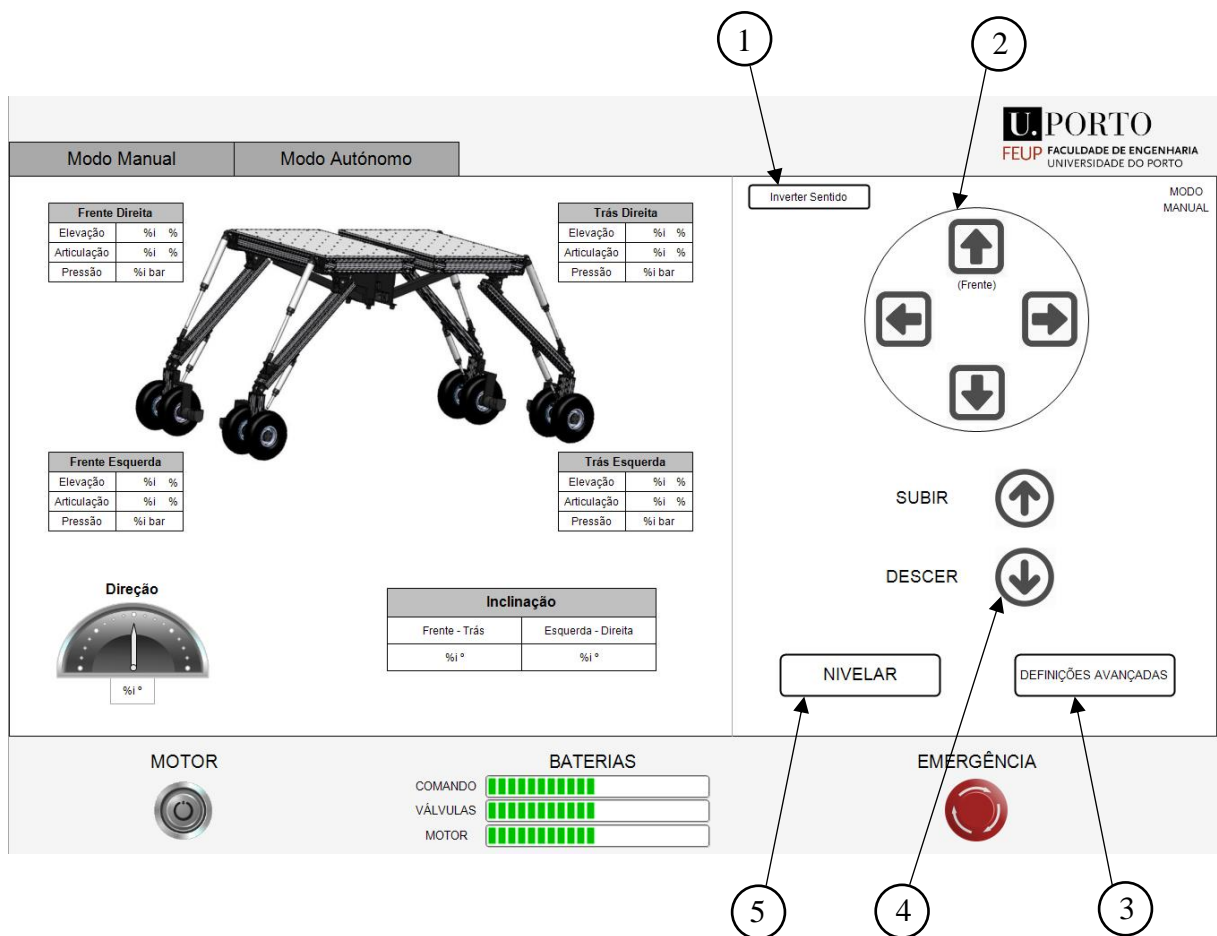


Figura 5.24 - Interface no modo manual

Tabela 6 - Legenda das funções da interface no modo manual

Número	Função
1	• Inverter o sentido dos comandos de 2
2	• Comando da translação e da direção
3	• Ir para modo de funcionamento manual avançado
4	• Comando da elevação do veículo
5	• Nivelar a superfície superior do veículo

A função 1 foi adicionada à interface porque, visto que a Mesa Hidráulica Móvel é simétrica e apresenta as mesmas funcionalidades para ambos os sentidos, não faz sentido que a navegação da mesa num dos sentidos seja efetuada em marcha-atrás. Quando se movia a mesa para o sentido definido como frente a navegação era simples, mas quando se pretendia mover no sentido contrário a tarefa exigia mais esforço mental pois os lados da direção eram os inversos. Para colmatar tais adversidades introduziu-se um botão que permite inverter o sentido da navegação como se vê na Figura 5.25.

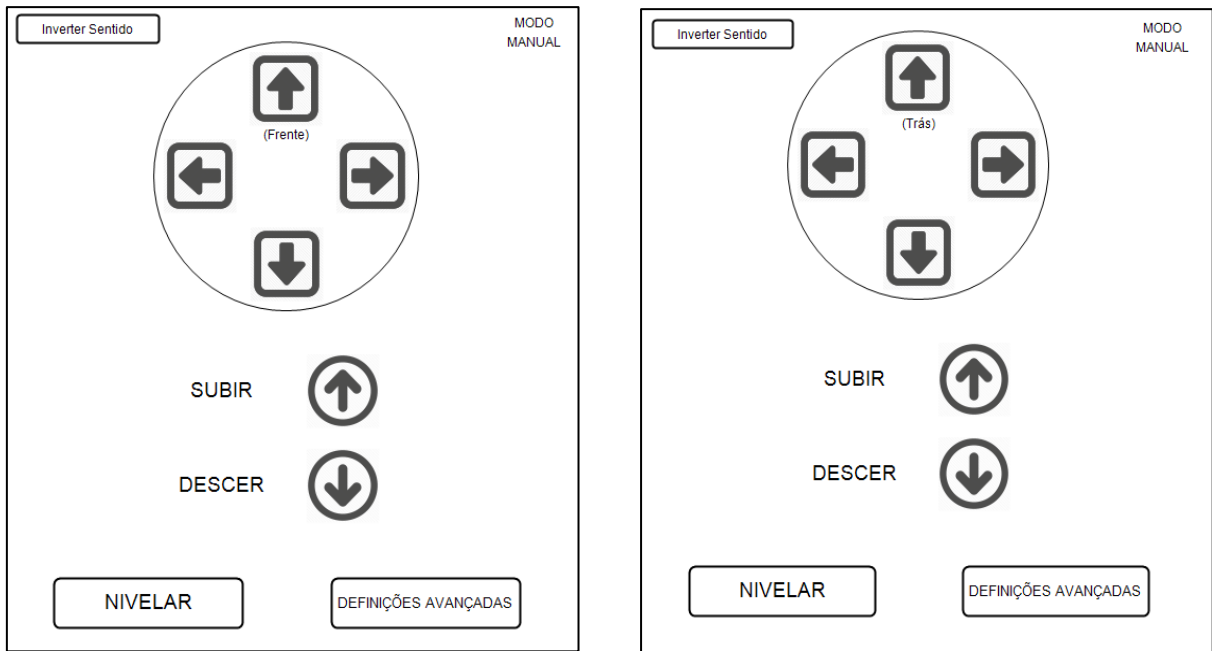


Figura 5.25 - Inversão do sentido de navegação

O comando da função 5, enquanto pressionado, ativa a nivelção corretiva do AGV no modo de funcionamento manual, pois esta tarefa se encontra, por norma, apenas ativada no modo autónomo.

#### 5.4.2 Modo manual avançado

O modo manual base resolve a maioria dos movimentos que se pretendem efetuar numa situação comum. Porém, quando é necessário ultrapassar um obstáculo mais desafiante, como um degrau, ou confirmar o funcionamento de todos os atuadores é necessário um modo de comandar cada atuador individualmente e, no primeiro caso, um operador experiente.

Foi para este género de situações que foi desenvolvido o modo manual avançado. Este modo permite não só a movimentação dos cilindros hidráulicos de elevação e de articulação individualmente, mas também o controlo da velocidade do motor, o comando do movimento de translação do veículo, a definição de qual o ângulo de viragem pretendido e o desbloqueamento das rodas quando o veículo está parado, que normalmente se encontram travadas por segurança.

Todas as funções deste modo estão enunciadas na Tabela 7 e podem ser associadas aos componentes exibidos na Figura 5.26

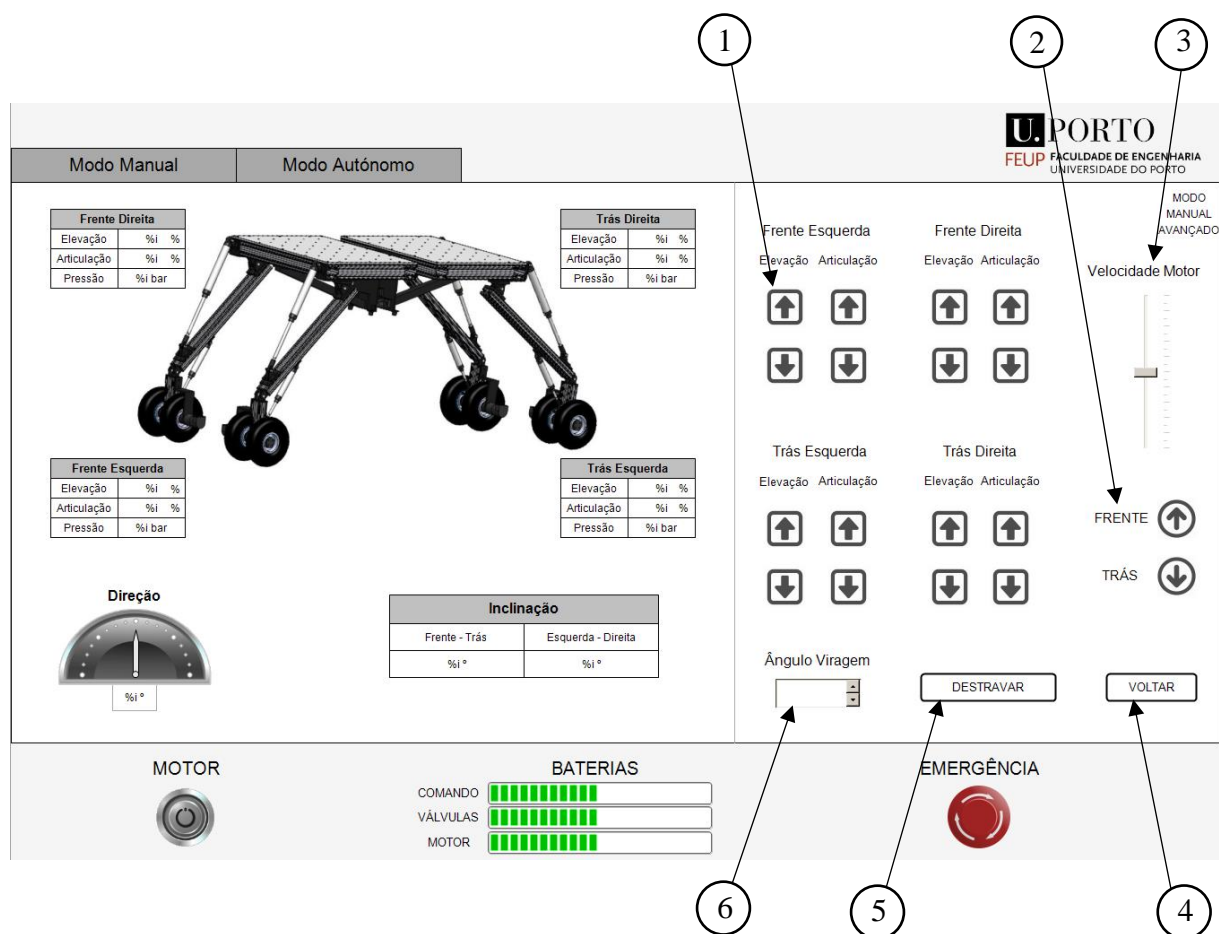


Figura 5.26 - Interface no modo manual avançado

Tabela 7 - Legenda das funções da interface no modo manual avançado

Número	Função
1	• Comando dos atuadores de elevação e articulação
2	• Comando da translação
3	• Controlo da velocidade do motor elétrico
4	• Voltar para modo de funcionamento manual base
5	• Desbloquear as rodas
6	• Definir ângulo de viragem

O controlo de velocidade no modo autónomo é efetuado automaticamente em função da distância ao colete refletor. Porém no modo manual avançado é possível fixar a velocidade do motor elétrico para o valor desejado.

A definição do ângulo de viragem é feita escolhendo o valor pretendido e o protótipo irá virar até atingir a mesma, parando de modo automático.

### 5.4.3 Modo autónomo

No modo autónomo não há necessidade de comandar nada, sendo a interface apenas para visualizar o estado das variáveis.

Neste modo é possível ver-se o estado de cada célula da grelha, tendo um código de cor identificando a gravidade da deteção e, aquando da identificação do colete refletor, aparece uma representação do mesmo indicando de que lado se encontra e qual a distância e o ângulo em que se encontra.

Pode-se ver o design deste modo de funcionamento na Figura 5.27 e relacionar com as funções descritas na Tabela 8.

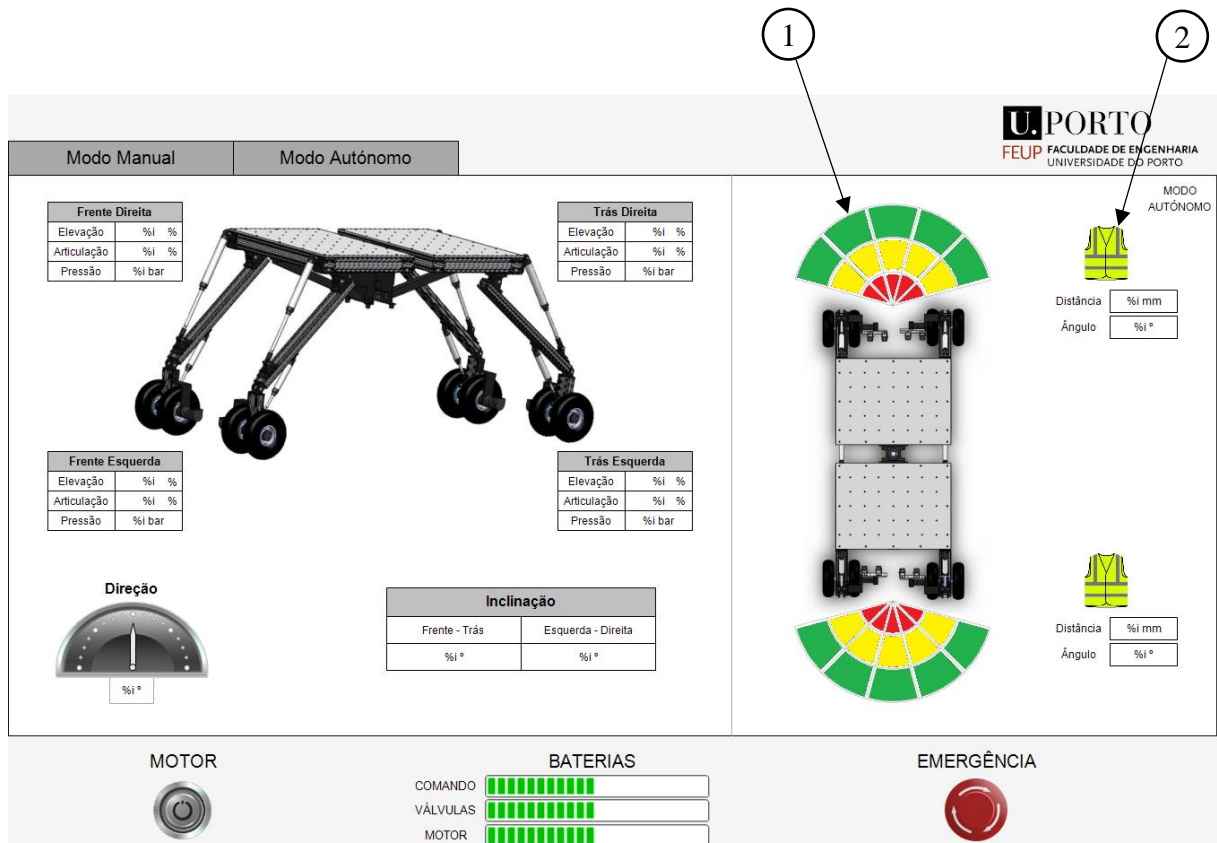


Figura 5.27 - Interface no modo autónomo

Tabela 8 - Legenda das funções da interface no modo autónomo

Número	Função
1	• Leitura dos sensores de proximidade
2	• Leitura do colete e sua distância e ângulo

A função 1 permite saber em que intervalo de distância se encontram os obstáculos, sendo que quanto mais perto se encontrar mais alarmante a cor da célula. Na Figura 5.28 pode-se ver a diferença entre a grelha sem nenhuma deteção e com todos os sensores com deteções muito próximas.

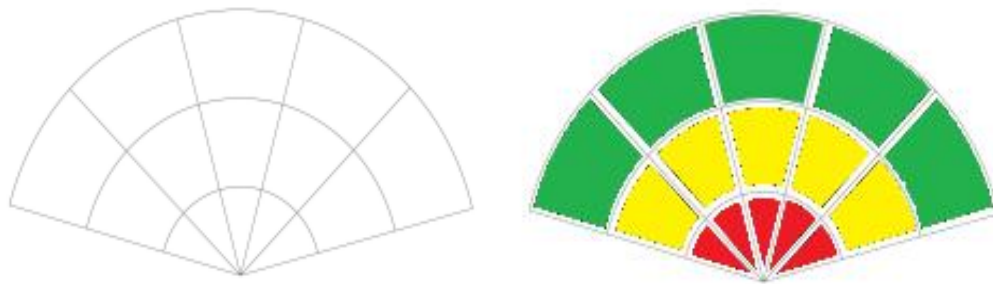


Figura 5.28 - Diferença entre grelha sem e com deteções, respetivamente

Na função 2 apenas um dos dois conjuntos de imagem e medidas se encontrará visível, pois o colete refletor ou se encontra atrás ou à frente, e existem ainda situações em que não estará nenhuma visível pois o colete não estará na linha de visão.

Devido a ser o único modo que depende da comunicação com as *Raspberry Pi*, este será o único que gera um estado de emergência aquando da falha de comunicação com as mesmas. Este estado de falha na comunicação ativa uma mensagem similar à da ativação da botoneira de emergência e pode ser observado na Figura 5.29.

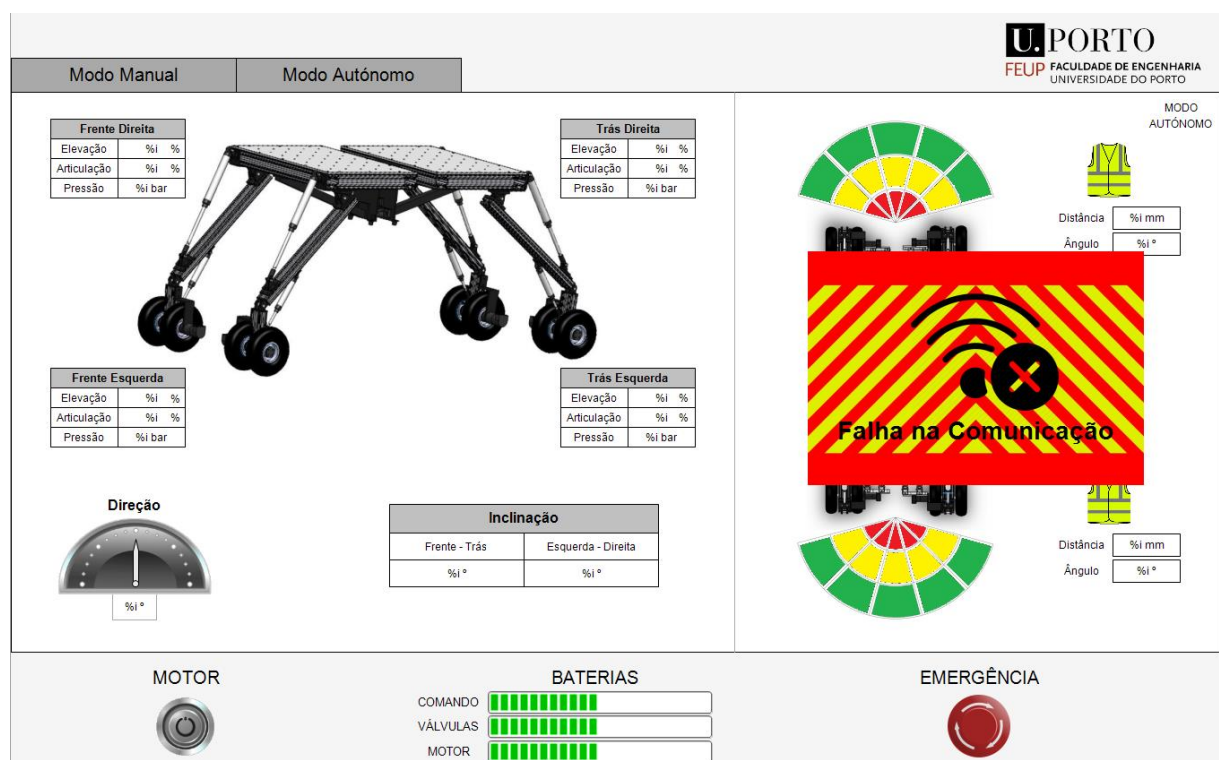


Figura 5.29 - Interface no estado de emergência por falha na comunicação



Tendo o design e as funções idealizados foi necessário configurar os componentes da interface. Devido ao elevado número de componentes apenas será exposto um para cada tipo de componente.

Inicialmente serão abordados os componentes que estão predefinidos no *software* e que a configuração é simples. Dentro destes iniciamos com o *Power switch* que foi usado para o comando do motor e permite comutar uma variável digital. Como se pode ver na Figura 5.30, tem de se definir qual a variável e se o botão é monoestável ou biestável (*tapper* ou *toggler*).

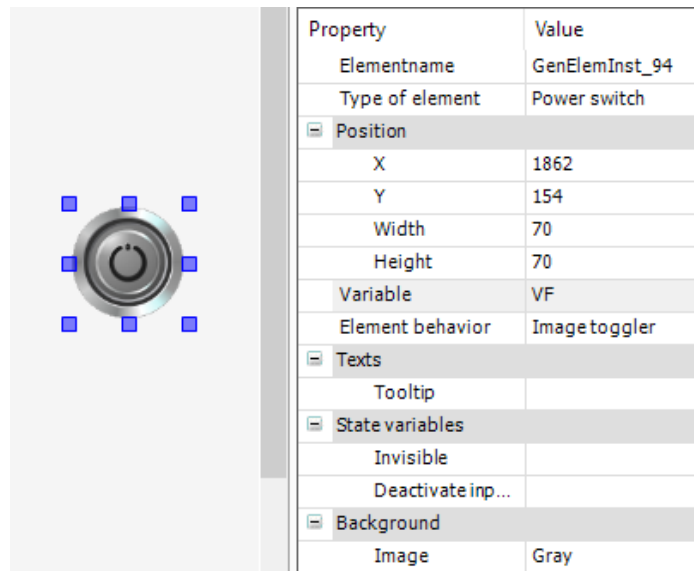


Figura 5.30 - Configuração de *Power switch*

Dos componentes base do sistema foram usadas duas formas diferentes de mostrar um valor analógico: o *Meter 180°* e o *Progress bar*. No primeiro tem de se definir, como se vê na Figura 5.31, qual a variável a exibir e os valores de início, fim e medida da escala. É ainda possível definir algumas questões estéticas como as cores, o fundo e o formato da seta.

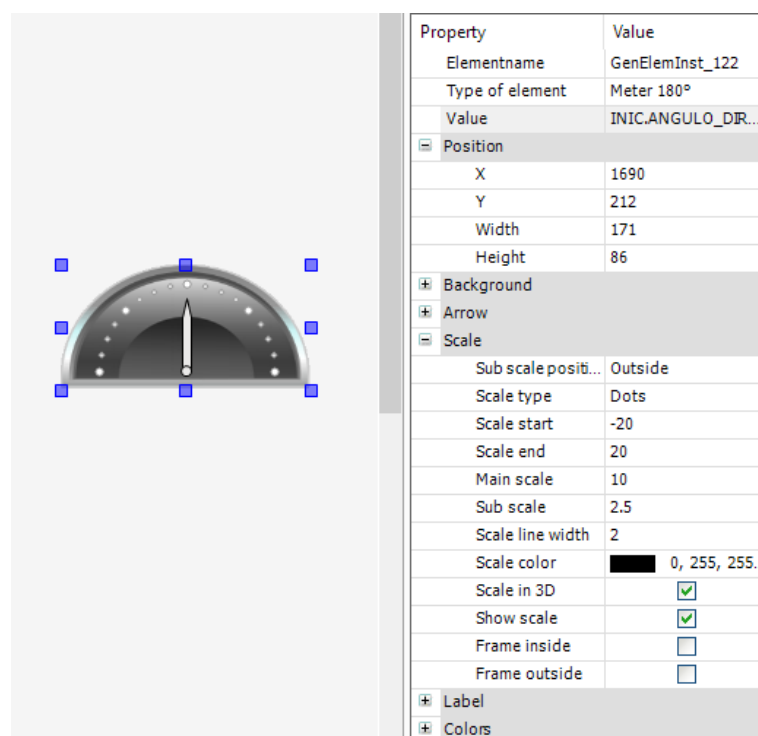


Figura 5.31 - Configuração de *Meter 180°*

O *Progress bar* que se encontra exposto na Figura 5.32 pode ser usado como uma barra contínua ou por blocos e tem de se definir qual a variável e o seu valor mínimo e máximo.

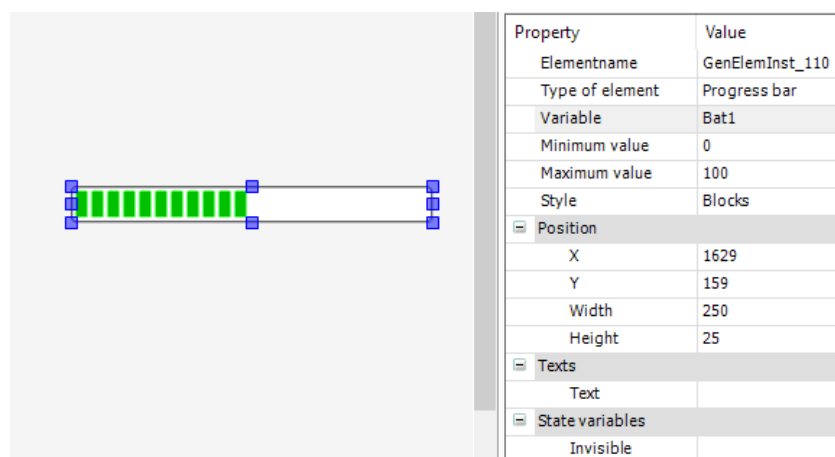


Figura 5.32 - Configuração do *Progress bar*

Para comandar variáveis analógicas foram também usados dois componentes diferentes, o *SpinControl* e o *Slider*. O *SpinControl* foi usado para definir qual o ângulo de viragem pois este componente permite alterar o valor em intervalos fixos entre valores positivos e negativos. Para o configurar tem de se introduzir qual a variável a ser controlada, o formato do valor (inteiro, real, número de casas decimais) que no caso presente será inteiro e qual o intervalo alterado por clique. A janela de configuração deste componente encontra-se representada na Figura 5.33.

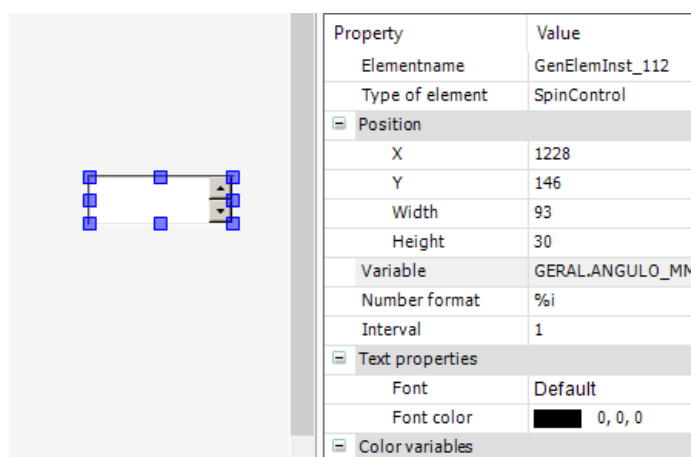


Figura 5.33 - Configuração do *SpinControl*

O último componente base do software usado foi o *Slider* e utilizou-se para definir a velocidade do motor elétrico de uma forma simples e intuitiva. Para configurar este componente tem de se seleccionar a variável a comandar, a direção e sentido da barra e o limite superior e inferior da escala.

Encontra-se na Figura 5.34 a janela de configuração deste elemento e os valores tomados para o mesmo.



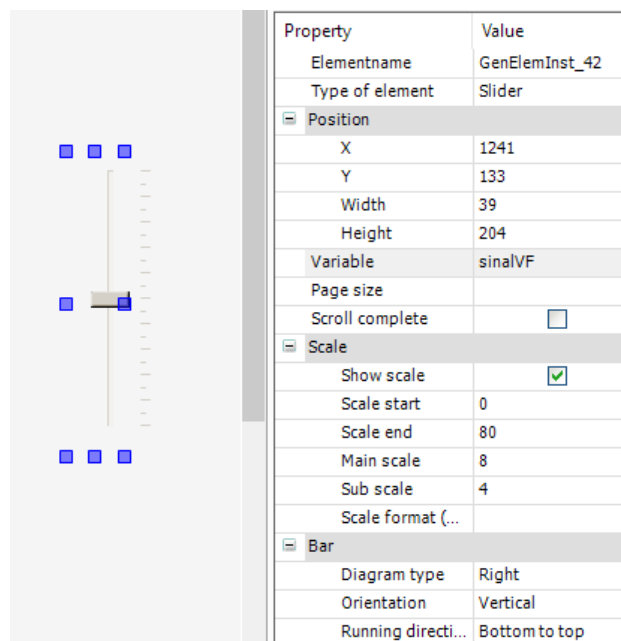


Figura 5.34 - Configuração do *Slider*

Embora a escolha de componentes não seja a mais vasta, a aplicação permite que sejam utilizadas formas e imagens como componentes de controlo. Esta opção revelou-se bastante útil, tendo sido usada para diversos fins.

Foram usados retângulos (*Rectangle*) para permitir visualizar os valores lidos pelos transdutores, sendo que para tal é necessário colocar na configuração do texto o formato do valor (%i significa inteiro) e selecionar a variável associada ao texto. É também boa prática colocar o componente como insensível a interações como se pode ver feito na Figura 5.35.

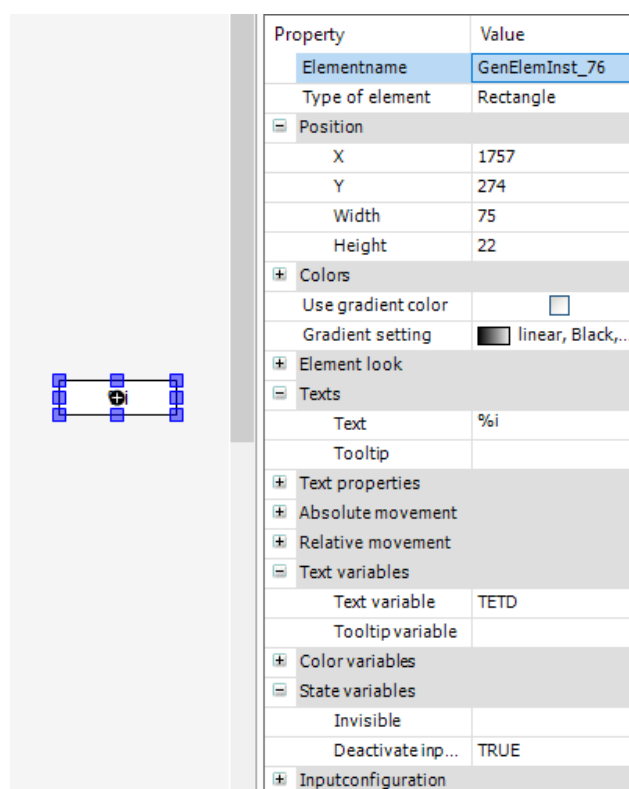


Figura 5.35 - Configuração do *Rectangle* para leitura de valores

Foram usados retângulos com os cantos arredondados (*Round rectangle*) e imagens (*Image*) para comando de diversas variáveis digitais, sendo isto possível utilizando a opção de *Inputconfiguration* que permite realizar ações quando se prime com o rato sobre estes componentes.

Dentro desta função tem-se a possibilidade de realizar ações quando se prime (*OnMouseDown*) e quando se solta (*OnMouseUp*) o botão esquerdo do rato sobre estes componentes, sendo possível observar estas configurações na Figura 5.36.

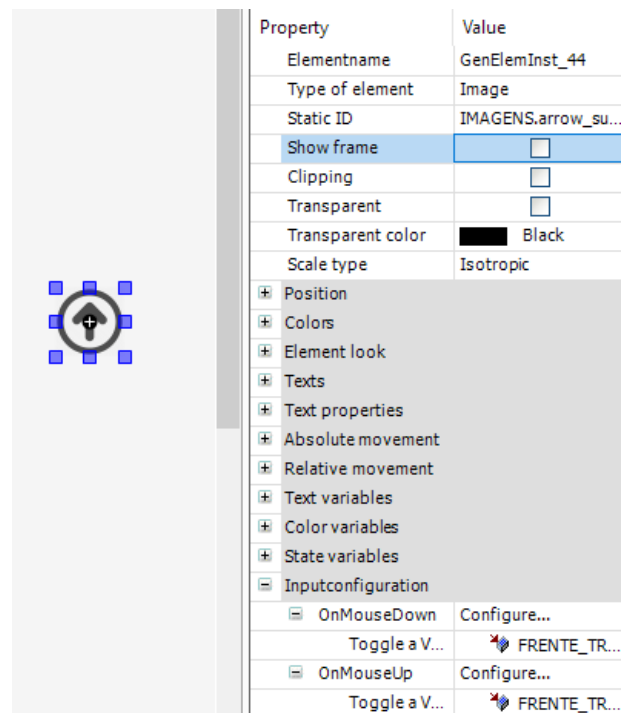


Figura 5.36 - Configuração de *Image*

As ações disponíveis são várias e não são multiplamente exclusivas, ou seja, é possível colocar mais do que uma ação no mesmo comando. A configuração das ações é realizada na janela visível na Figura 5.37.

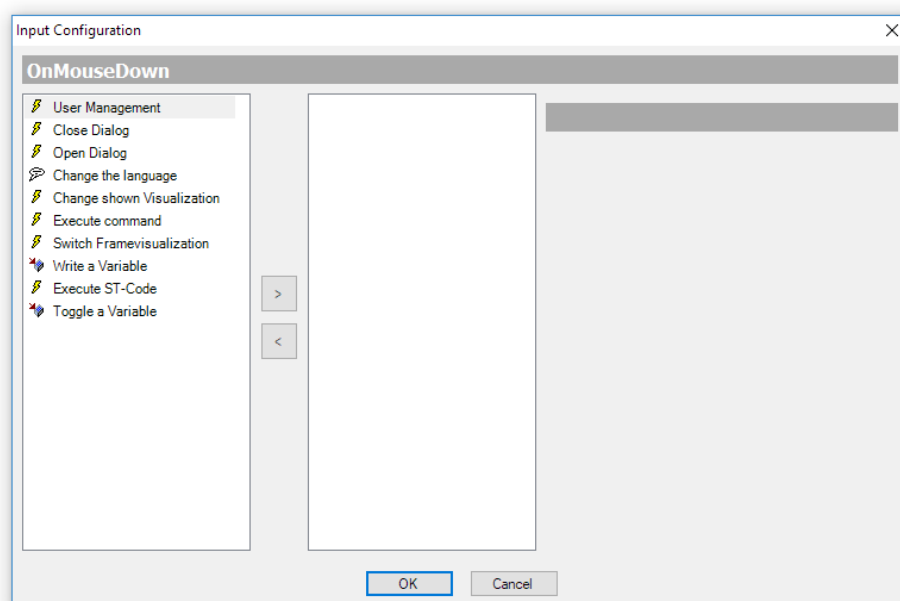


Figura 5.37 - Configuração de *OnMouseDown*

Dentro de todas as ações possíveis apenas duas foram utilizadas, estas foram a de trocar o valor de uma variável (*Toggle a Variable*), ou seja, se estiver em zero passar para um e vice-versa e a de mudar a visualização (*Switch Framevisualization*) que permite trocar as janelas do modo de funcionamento na mesma interface.

Na ação de alterar o valor de uma variável é apenas necessário configurar qual a variável a ser comutada, sendo possível ver na Figura 5.38 este processo. Usou-se esta ação apenas no clique do rato quando se pretendia que a imagem agisse como um botão biestável e no clique e levantar do rato quando se queria um botão monoestável.

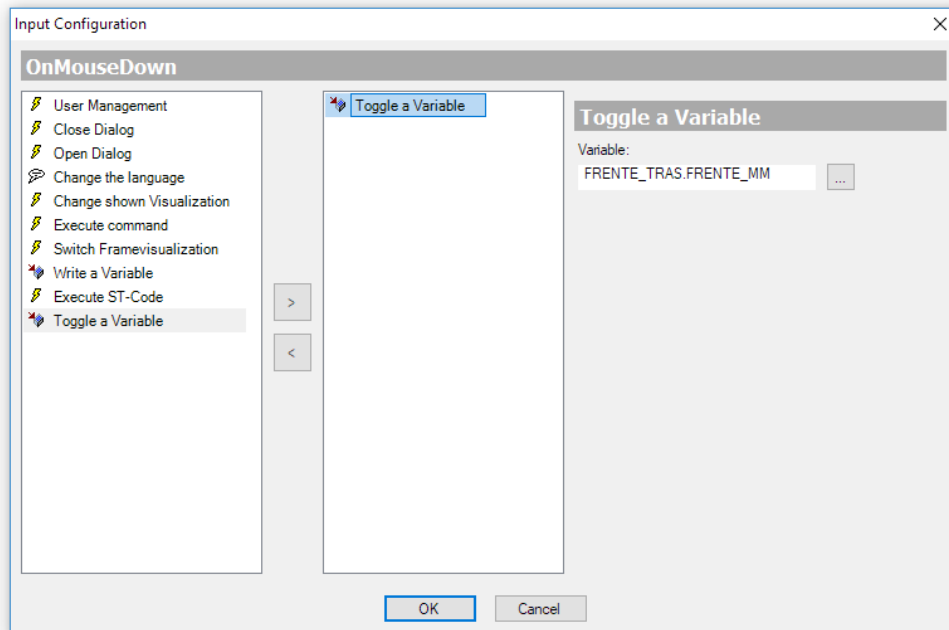


Figura 5.38 - Configuração de *Toggle a Variable*

A alteração de visualização foi usada para comutar entre os *frames* dos modos de funcionamento. Esta ação permite definir qual o *frame* que se pretende que apareça na interface pelo método mostrado na Figura 5.39.

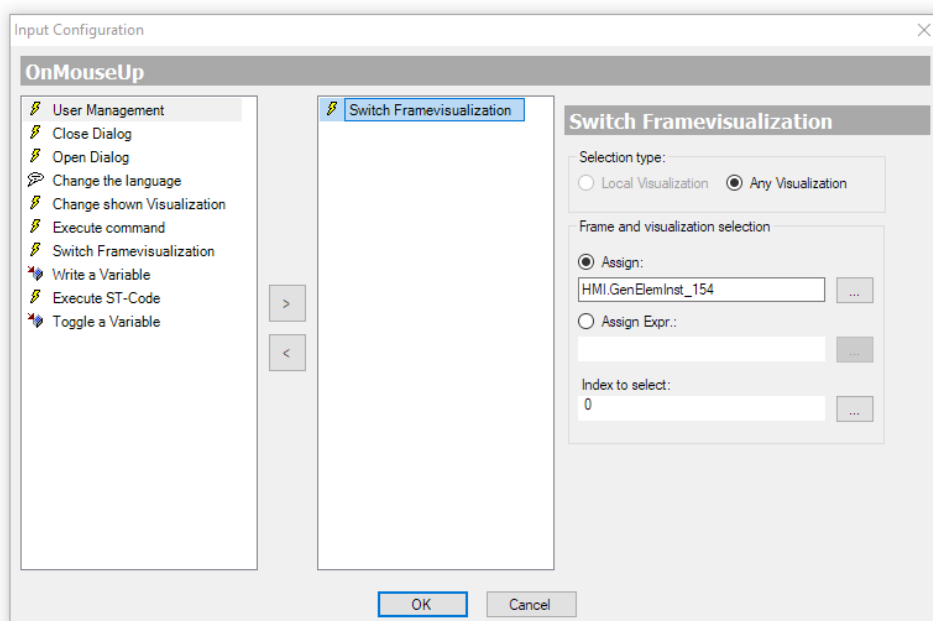


Figura 5.39 - Configuração de *Switch Framevisualization*

Para configurar esta ação é preciso escolher qual o elemento de visualização de *frames* (*Frame*) e qual o *frame* a mostrar, sendo esta escolha feita pelo índice que é previamente definido na configuração do elemento *Frame*, como se vê na Figura 5.40.

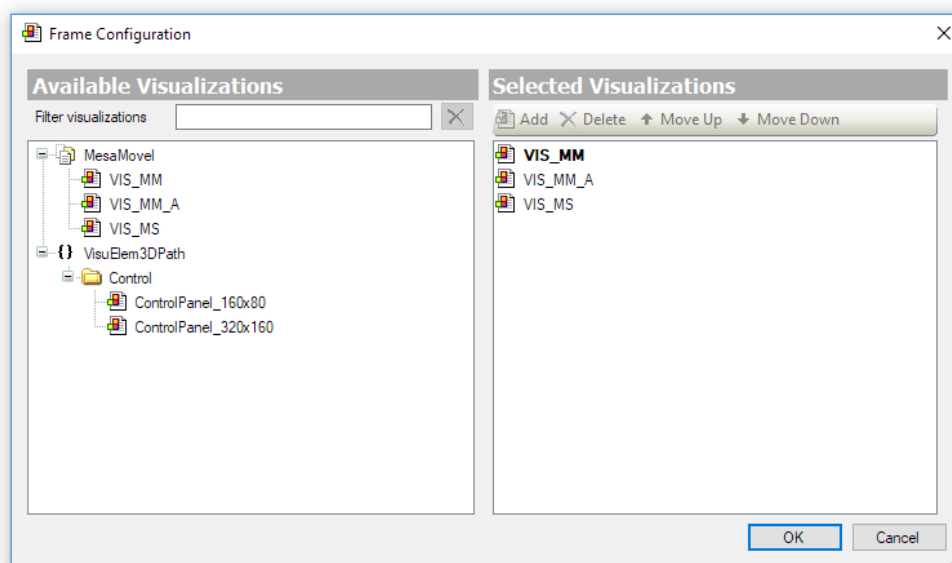


Figura 5.40 - Configuração de *Frame*

Por fim, utilizou-se a possibilidade de colocar os diversos componentes invisíveis através da associação a uma variável. Isto foi útil, por exemplo, na implementação dos alarmes de emergência que se mantêm invisíveis quando não se encontra ativada a emergência, tal como é notório na Figura 5.41.



Figura 5.41 - Configuração invisibilidade de *Image*

Deste modo deu-se por concluído o desenvolvimento da interface e, por consequente, do projeto.

## 6 Conclusão

Atingido o final deste trabalho é possível afirmar que existiu uma evolução significativa na tecnologia instalada na Mesa Hidráulica Móvel e que os objetivos foram cumpridos, tendo ficado o veículo a funcionar como previsto tanto no modo manual como no modo autónomo.

Foi realizada uma atualização de diversos componentes que permitiu transformar o que era um protótipo tecnologicamente ultrapassado, com componentes antigos, pouco capazes em relação aos atuais e alguns deles inutilizáveis, num veículo funcional e com potencial para novos desenvolvimentos.

Foi cumprido o objetivo de instalar um modo de navegação com mais potencial, tendo sido trocado um modo funcional pouco flexível e sem espaço para evolução, por um modo com alta flexibilidade e enorme potencial para aumento de funcionalidades.

Em paralelo com o *hardware*, foi desenvolvido *software* que acomodasse esta evolução. Foi criado um programa de controlo com funcionalidades idênticas ao anterior para o novo PLC, mas atualizado para o novo modo de navegação e um programa no microcomputador capaz de executar o processamento da imagem.

Foi ainda alterado o modo de comunicação de *Bluetooth* para *Wi-Fi*, aumentando a distância a que se pode encontrar o computador que controla a Mesa Hidráulica Móvel, tendo sido desenvolvida também uma nova interface, simples e intuitiva, para o comando da mesma.

### 6.1 Trabalhos futuros

Dada como terminada a quarta fase, é ainda notória uma grande potencialidade para evolução deste protótipo. Ao longo deste projeto foram surgindo diversas ideias que, devido a limites temporais, não foram exequíveis no decorrer deste trabalho.

Ficam as seguintes ideias como sugestões para implementar numa futura fase de desenvolvimento da Mesa Hidráulica Móvel:

- Criação de um circuito elétrico que possibilite o carregamento simples e simultâneo de todas as baterias;
- Desenvolvimento da capacidade de fazer o veículo caminhar e ultrapassar pequenas escadas de modo autónomo;
- Desenvolvimento da utilização da visão como meio de navegação adicionando funcionalidades como a deteção de gestos ou identificação da sua posição por triangulação.

## Bibliografia

- [1] - Teixeira, Tiago. 2009. “Mesa Hidráulica Móvel” Dissertação de Mestrado. Faculdade de Engenharia da Universidade do Porto.
- [2] - Ribeiro, Manuel. 2010. “Controlo De Movimento Em Mesa Hidráulica Móvel” Dissertação de Mestrado. Faculdade de Engenharia da Universidade do Porto.
- [3] - “PSI 1000/2000 SERIES - User Manual” PDF. HQ-Power. Último acesso: agosto 2018, <https://www.velleman.eu/downloads/2/psi10002000seriesgbnlfresdit.pdf>
- [4] - Transbotics Corporation. "Guidance and Navigation for AGV." AGV and AGC Automatic Guided Vehicle Manufacturer. Último acesso: agosto 2018, <https://www.transbotics.com/learning-center/guidance-navigation>.
- [5] - "AGVS Basic Information." Savant Automation. Agosto 2013. Último acesso: agosto 2018, <http://www.agvsystems.com/agvs-basics/basics-agvs/>.
- [6] - “Raspberry Pi 3 Model B” PDF. RS. Último acesso: agosto 2018, <https://docs-apac.rs-online.com/webdocs/1529/0900766b81529c6c.pdf>
- [7] - “Raspberry Pi 3 Model B” PDF. Raspberry Pi. Último acesso: agosto 2018, <https://static.raspberrypi.org/files/product-briefs/Raspberry-Pi-Model-Bplus-Product-Brief.pdf>
- [8] - “Raspberry Pi Camera Module” PDF. RS. Último acesso: agosto 2018, <https://cdn.sparkfun.com/datasheets/Dev/RaspberryPi/RPiCamMod2.pdf>
- [9] - “Ultrasonic Ranging Module HC - SR04” PDF. ElecFreaks. Último acesso: agosto 2018, <https://www.mouser.com/ds/2/813/HCSR04-1022824.pdf>
- [10] - Matt. "Simple Guide to the Raspberry Pi GPIO Header and Pins." Raspberry Pi Spy. junho, 2018. Último acesso: agosto 2018, <https://www.raspberrypi-spy.co.uk/2012/06/simple-guide-to-the-rpi-gpio-header-and-pins/#prettyPhoto/0/>
- [11] - “Product Data Sheet - TM251MESC” PDF. Schneider Electric. Último acesso: agosto 2018, <https://www.schneider-electric.com/en/product/download-pdf/TM251MESC>
- [12] - “Product Data Sheet - TM3AI8G” PDF. Schneider Electric. Último acesso: agosto 2018, <https://www.schneider-electric.com/en/product/download-pdf/TM3AI8G>
- [13] - “Product Data Sheet - TM3DM24R” PDF. Schneider Electric. Último acesso: agosto 2018, <https://www.schneider-electric.com/en/product/download-pdf/TM3DM24R>
- [14] - “Product Data Sheet - TM3AM6G” PDF. Schneider Electric. Último acesso: agosto 2018, <https://www.schneider-electric.com/en/product/download-pdf/TM3AM6G>
- [15] - “Product Data Sheet - TM3DQ16RG” PDF. Schneider Electric. Último acesso: agosto 2018, <https://www.schneider-electric.com/en/product/download-pdf/TM3DQ16RG>
- [16] - “Product Data Sheet - TM4ES4” PDF. Schneider Electric. Último acesso: agosto 2018, <https://www.schneider-electric.com/en/product/download-pdf/TM4ES4>
- [17] - “Valve-Regulated Lead Acid Batteries: Individual Data Sheet LC-R1233P”. PDF. Panasonic. Último acesso: agosto 2018, <http://www.clearpowersolutions.com/uploads/lcr1233p.pdf>
- [18] - “SYSDRIVE 3G3JV - User's Manual” PDF. OMRON, 1999. Último acesso: agosto 2018, [https://www.limasoft.cz/omron/pdf/3G3JV\\_en\\_manual.pdf](https://www.limasoft.cz/omron/pdf/3G3JV_en_manual.pdf)
- [19] - "300Mbps Wireless N Wall-Plate Access Point EAP115-Wall." TP-Link. Último acesso: agosto 2018, [https://www.tp-link.com/uk/products/details/cat-5694\\_EAP115-Wall.html#overview](https://www.tp-link.com/uk/products/details/cat-5694_EAP115-Wall.html#overview)

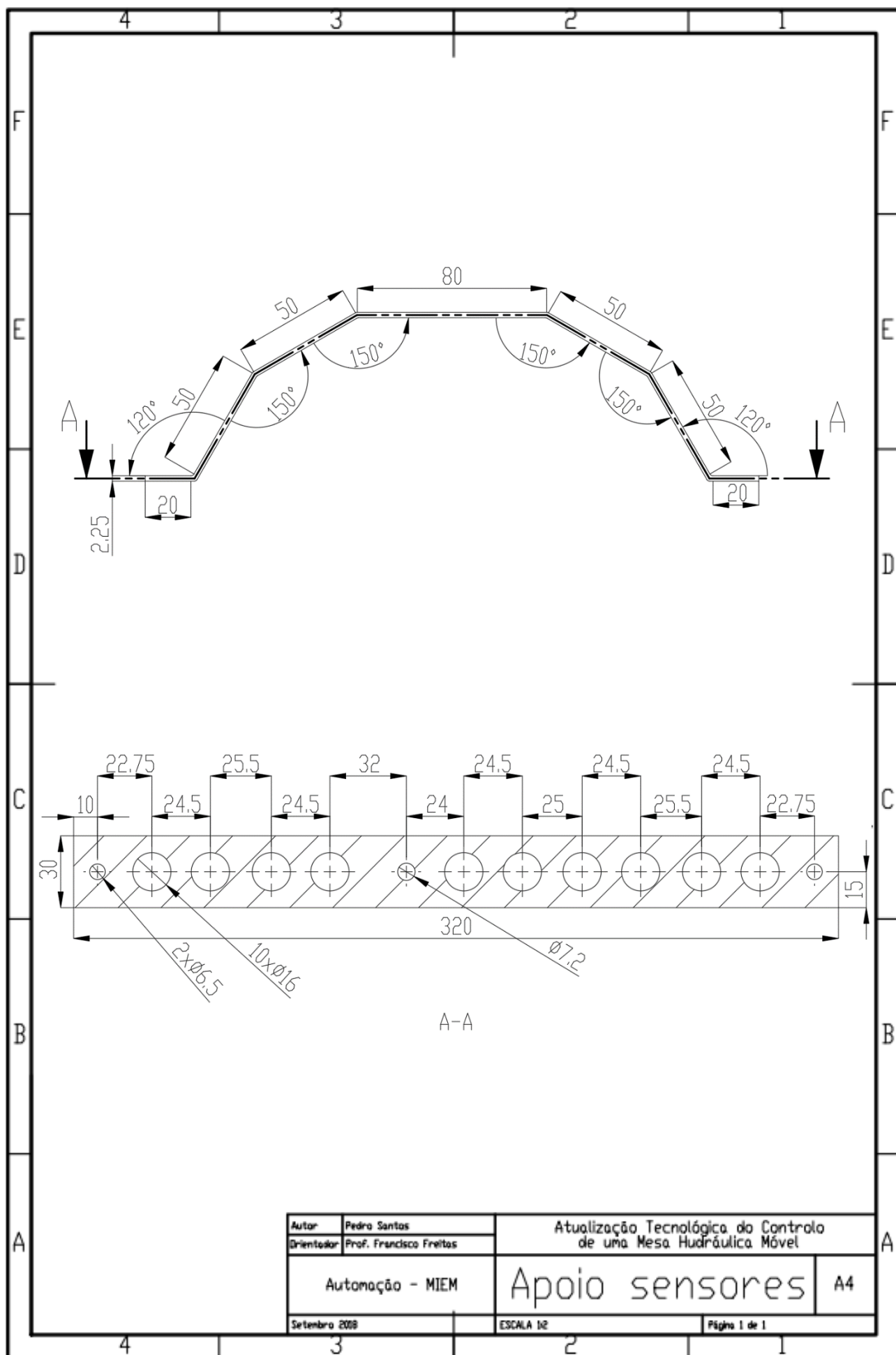
[20] - “PoE Injector TL-POE150S” TP-Link. Último acesso: agosto 2018, [https://www.tp-link.com/us/products/details/cat-43\\_TL-POE150S.html](https://www.tp-link.com/us/products/details/cat-43_TL-POE150S.html)

[21] - Bocafoli, Francisco. Física e Vestibular. Último acesso: agosto 2018, [http://fisicaevestibular.com.br/novo/wp-content/uploads/universidades-2016/uerj/o\\_819c0c5dcb15134d.html](http://fisicaevestibular.com.br/novo/wp-content/uploads/universidades-2016/uerj/o_819c0c5dcb15134d.html)





## **ANEXO A: Desenho de construção**



## ANEXO B: Tabela de ligações ao autômato

Legenda:

<u>E</u> XX_X ou <u>A</u> XX_X ou <u>M</u> XX_X	Elevação, articulação ou motor, respetivamente.
X <u>F</u> X_X ou X <u>T</u> X_X	Frente ou traseira do veículo, respetivamente.
XX <u>D</u> _X ou XX <u>E</u> _X	Direita ou esquerda do veículo, respetivamente.
XXX_ <u>a</u> ou XXX_ <u>b</u>	Solenóide a ou b, respetivamente.

Módulo	I/O	Função	Ligações
Módulo Base TM251MESC	Alimentação	24 VDC - comando	Branco
		0 VDC	Azul
		Terra	Amarelo/verde
	Ethernet.1	Comunicação com módulo switch	RJ45
	Ethernet.2		
	CANopen		
	Serial		
	Mini USB		
Switch TM4ES4 Módulo 0	Ethernet.1	Comunicação com módulo base	RJ45
	Ethernet.2	Comunicação com Raspberry Pi frente	RJ45
	Ethernet.3	Comunicação com Raspberry Pi trás	RJ45
	Ethernet.4	Comunicação com Access Point	RJ45

Entradas e saídas digitais TM3DM24R Módulo 1	ID1.0		
	ID1.1		
	ID1.2		
	ID1.3		
	ID1.4		
	ID1.5		
	ID1.6		
	ID1.7		
	ID1.8		
	ID1.9		
	ID1.10		
	ID1.11		
	ID1.12		
	ID1.13		
	ID1.14		
	ID1.15	Botoneira de emergência	Preto
	COM0	0 VDC	Azul
	OD1.0		
	OD1.1	Contrapressão	Cinzeno/rosa
	OD1.2	Direção esquerda	Amarelo/cinzeno
	OD1.3	Direção direita	Amarelo/branco
	COM1	24 VDC - potência	Castanho
	NC		
	OD1.4		
	OD1.5		
	OD1.6		
	OD1.7		Branco/rosa
	COM2	Variador de frequência	Branco/azul
Entradas e saídas digitais TM3AM6G Módulo 2	Alimentação	24 VDC - comando	Branco
		0 VDC	Azul
	OA2.0+	Sinal comando variador de frequência	Castanho/verde
	OA2.0-		Castanho/cinzeno
	OA2.1+		
	OA2.1-		
	IA2.0+		
	IA2.0-		
	NC		
	NC		
	IA2.1+		
	IA2.1-		
	NC		
	IA2.2+	Inclinômetro X	Amarelo
	IA2.2-	0 VDC	Azul
	NC		
	IA2.3+	Inclinômetro Y	Verde
	IA2.3-	0 VDC	Azul
	NC		

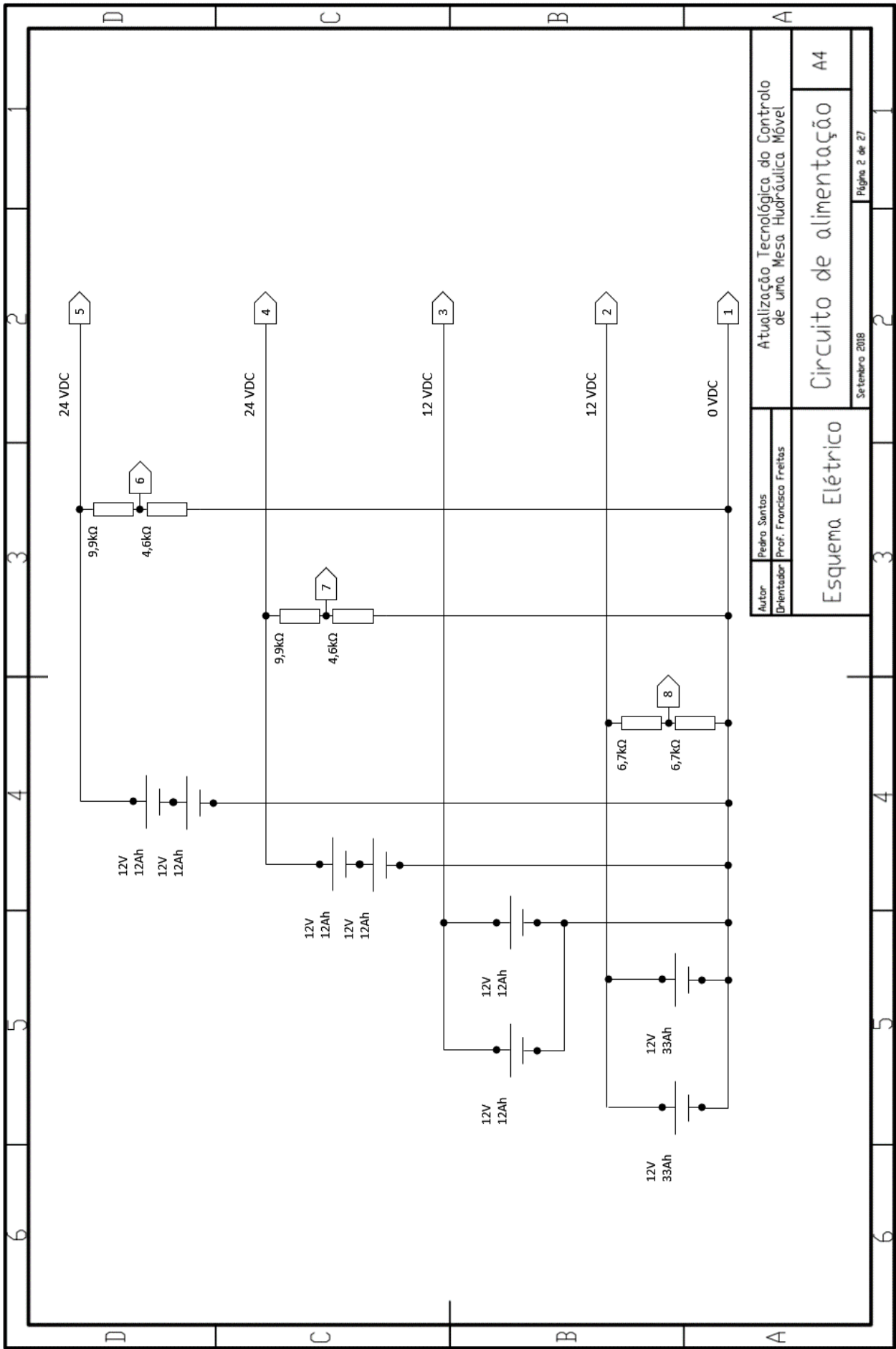
Oito Entradas analógicas TM3AI8G Módulo 3	Alimentação	24 VDC - comando	Branco
		0 VDC	Azul
	NC		
	IA3.0+	Transdutor posição da junta EFD	Vermelho
	IA3.0-	0 VDC	Azul
	IA3.1+	Transdutor posição da junta AFD	Roxo
	IA3.1-	0 VDC	Azul
	IA3.2+	Transdutor posição da junta EFE	Cinza/rosa
	IA3.2-	0 VDC	Azul
	IA3.3+	Transdutor posição da junta AFE	Preto
	IA3.3-	0 VDC	Azul
	IA3.4+	Transdutor posição da junta ETD	Verde
	IA3.4-	0 VDC	Azul
	IA3.5+	Transdutor posição da junta ATD	Amarelo
	IA3.5-	0 VDC	Azul
	IA3.6+	Transdutor posição da junta ETE	Branco
	IA3.6-	0 VDC	Azul
	IA3.7+	Transdutor posição da junta ATE	Rosa
	IA3.7-	0 VDC	Azul
Oito Entradas analógicas TM3AI8G Módulo 4	Alimentação	24 VDC - comando	Branco
		0 VDC	Azul
	NC		
	IA4.0+	Carga baterias 24 VDC comando	Branco
	IA4.0-	0 VDC	Azul
	IA4.1+	Carga baterias 24 VDC potência	Rosa
	IA4.1-	0 VDC	Azul
	IA4.2+	Carga baterias 12 VDC potência	Amarelo
	IA4.2-	0 VDC	Azul
	IA4.3+	Transdutor posição da direção	Castanho
	IA4.3-	0 VDC	Azul
	IA4.4+	Transdutor pressão FD	Castanho/amarelo
	IA4.4-	0 VDC	Azul
	IA4.5+	Transdutor pressão FE	Branco/verde
	IA4.5-	0 VDC	Azul
	IA4.6+	Transdutor pressão TD	Azul/vermelho
	IA4.6-	0 VDC	Azul
	IA4.7+	Transdutor pressão TE	Branco/amarelo
	IA4.7-	0 VDC	Azul

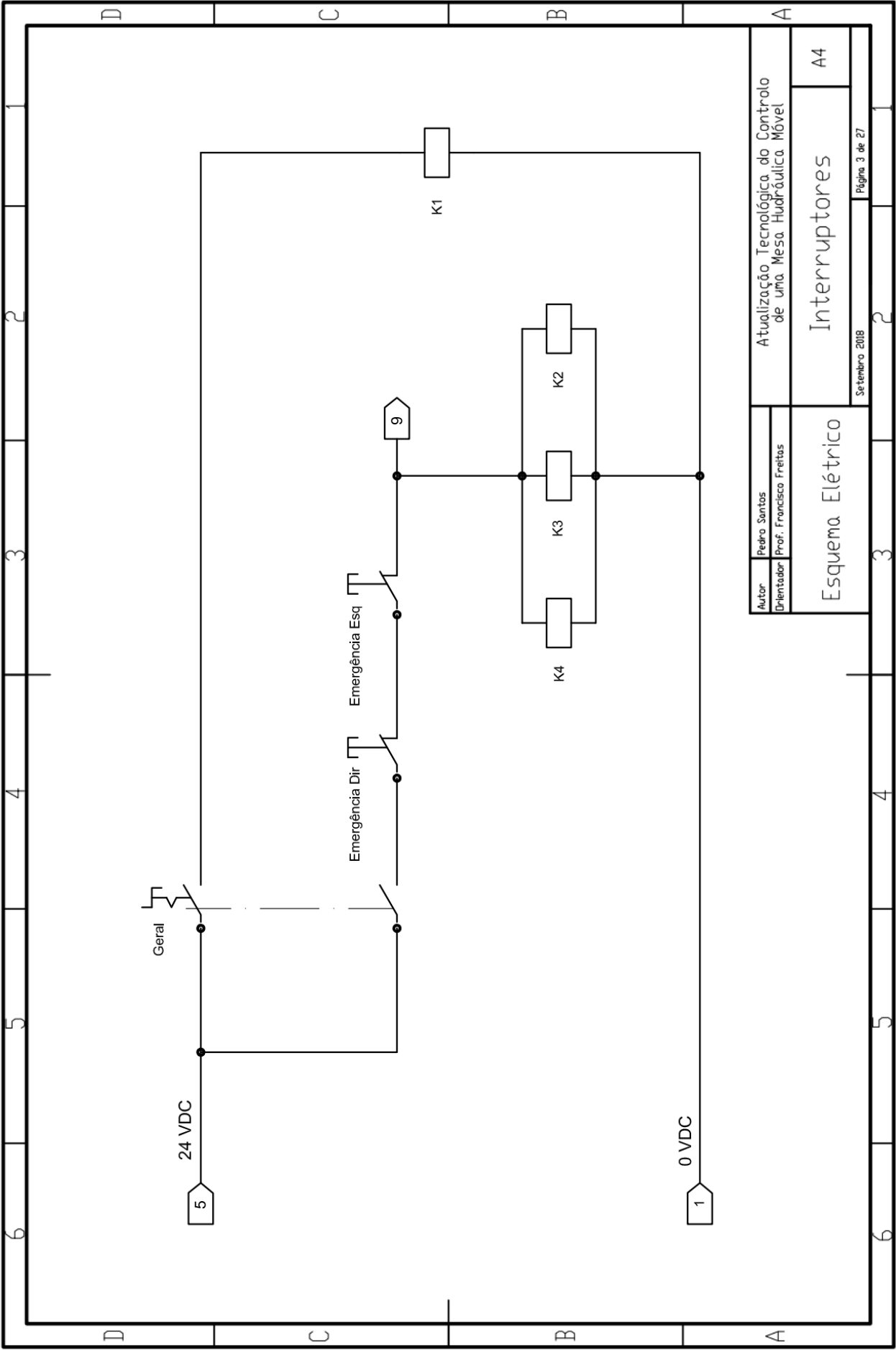
Dezasseis saídas digitais por relé TM3DQ16RG Módulo 5	OD5.0	Solenóide EFD_b	Castanho/cinza
	OD5.1	Solenóide EFD_a	Castanho/amarelo
	OD5.2	Solenóide AFD_b	Castanho/roza
	OD5.3	Solenóide AFD_a	Preto/castanho
	OD5.4	Solenóide MFD_b	Preto/amarelo
	OD5.5	Solenóide MFD_a	Preto/verde
	OD5.6	Solenóide FD travão	Preto/branco
	OD5.7	Solenóide FD carga	Castanho/verde
	COM0	24 VDC - potência	Castanho
	COM0	24 VDC - potência	Castanho
	OD5.8	Solenóide EFE_b	Verde/cinza
	OD5.9	Solenóide EFE_a	Verde/roza
	OD5.10	Solenóide AFE_b	Verde/branco
	OD5.11	Solenóide AFE_a	Rosa/cinza
	OD5.12	Solenóide MFE_b	Rosa/amarelo
	OD5.13	Solenóide MFE_a	Rosa/branco
	OD5.14	Solenóide FE travão	Vermelho/verde
	OD5.15	Solenóide FE carga	Vermelho/branco
	COM1	24 VDC - potência	Castanho
	COM1	24 VDC - potência	Castanho
Dezasseis saídas digitais por relé TM3DQ16RG Módulo 6	OD6.0	Solenóide ETD_b	Vermelho/castanho
	OD6.1	Solenóide ETD_a	Vermelho/amarelo
	OD6.2	Solenóide ATD_b	Branco
	OD6.3	Solenóide ATD_a	Cinza
	OD6.4	Solenóide MTD_b	Amarelo
	OD6.5	Solenóide MTD_a	Rosa
	OD6.6	Solenóide TD travão	Azul
	OD6.7	Solenóide TD carga	Roxo
	COM0	24 VDC - potência	Castanho
	COM0	24 VDC - potência	Castanho
	OD6.8	Solenóide ETE_b	Verde
	OD6.9	Solenóide ETE_a	Preto
	OD6.10	Solenóide ATE_b	Castanho
	OD6.11	Solenóide ATE_a	Vermelho
	OD6.12	Solenóide MTE_b	Azul/verde
	OD6.13	Solenóide MTE_a	Azul/vermelho
	OD6.14	Solenóide TE travão	Azul/castanho
	OD6.15	Solenóide TE carga	Azul/amarelo
	COM1	24 VDC - potência	Castanho
	COM1	24 VDC - potência	Castanho

## **ANEXO C: Esquemas elétricos**

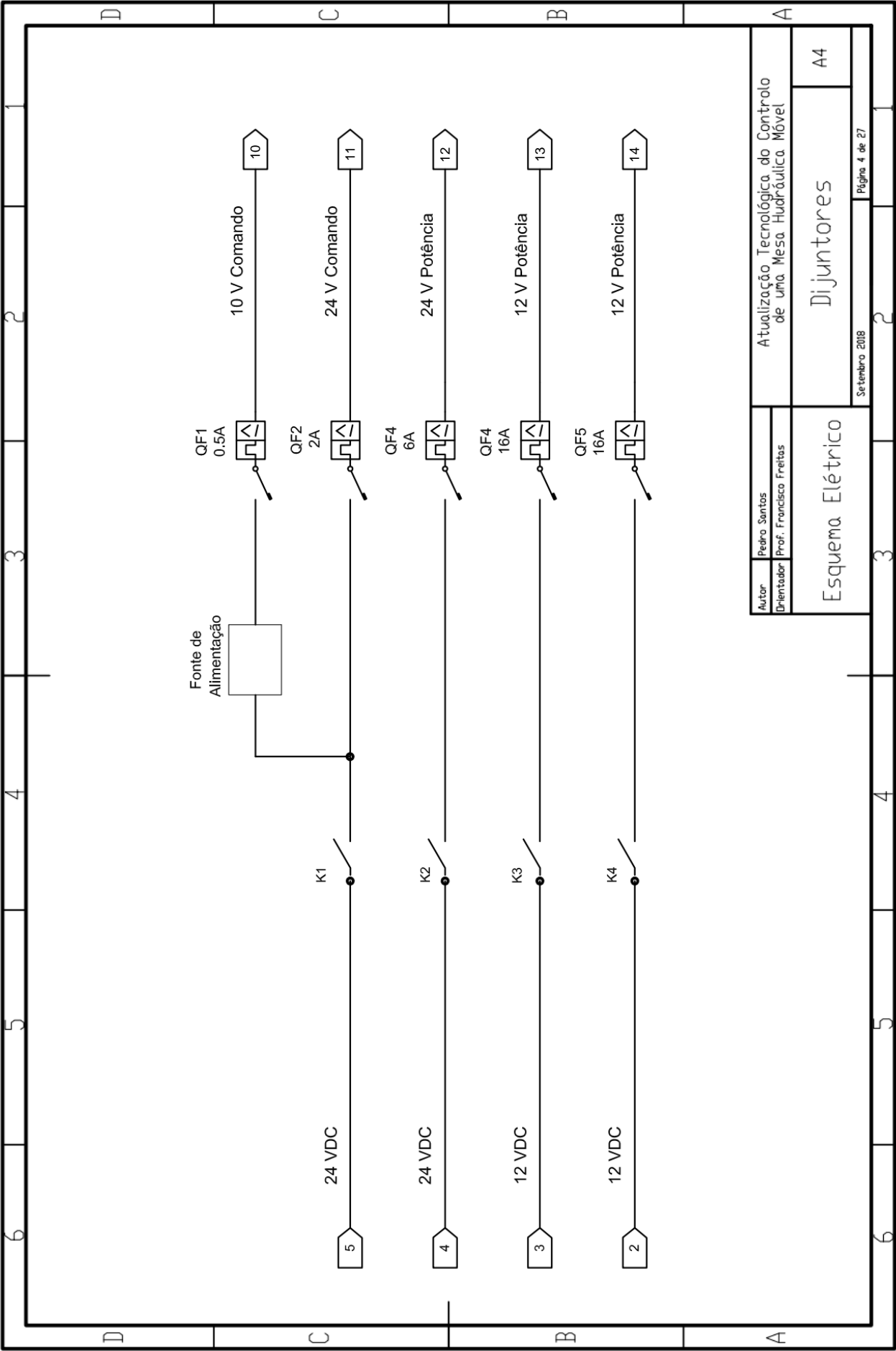


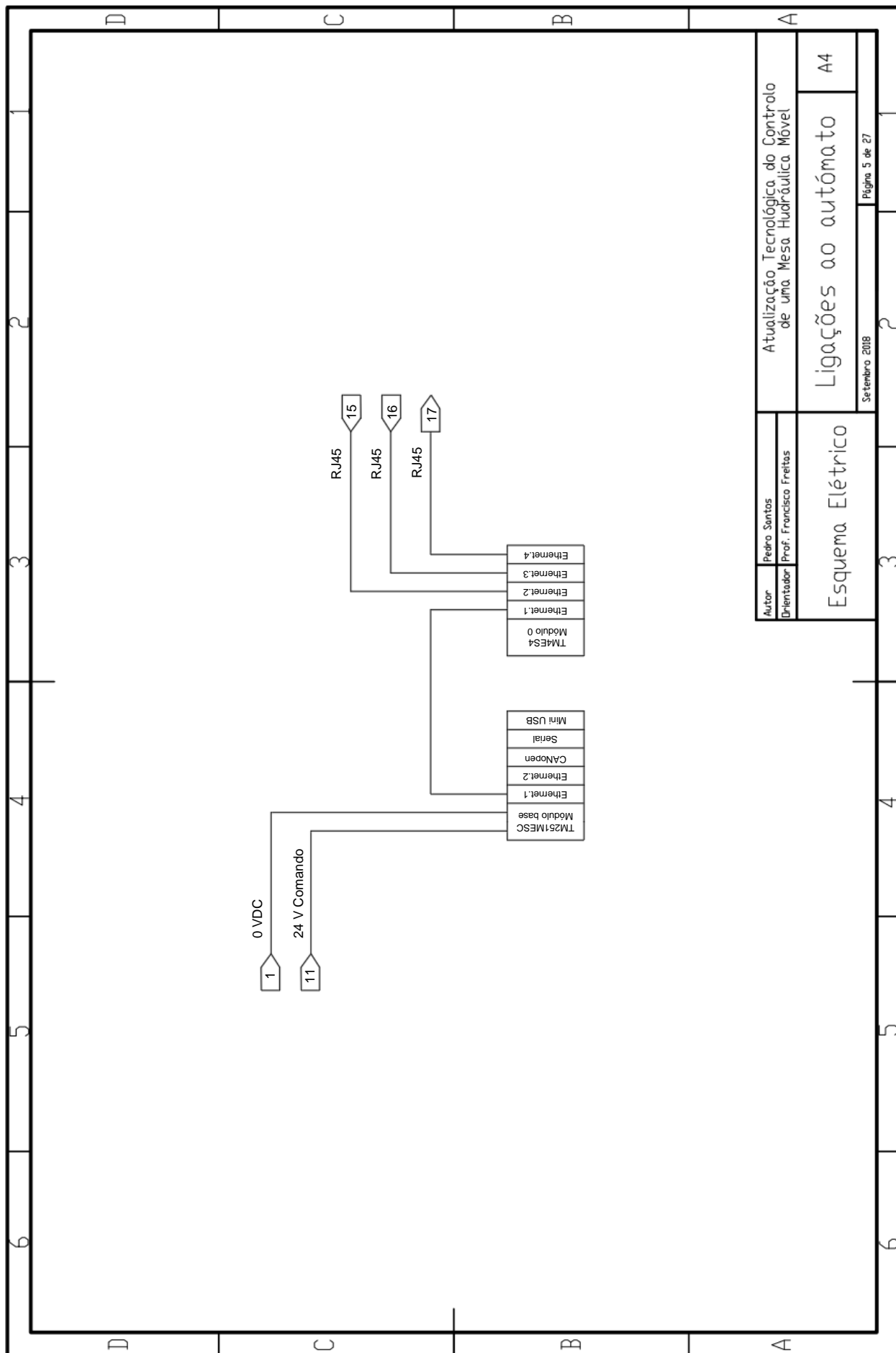




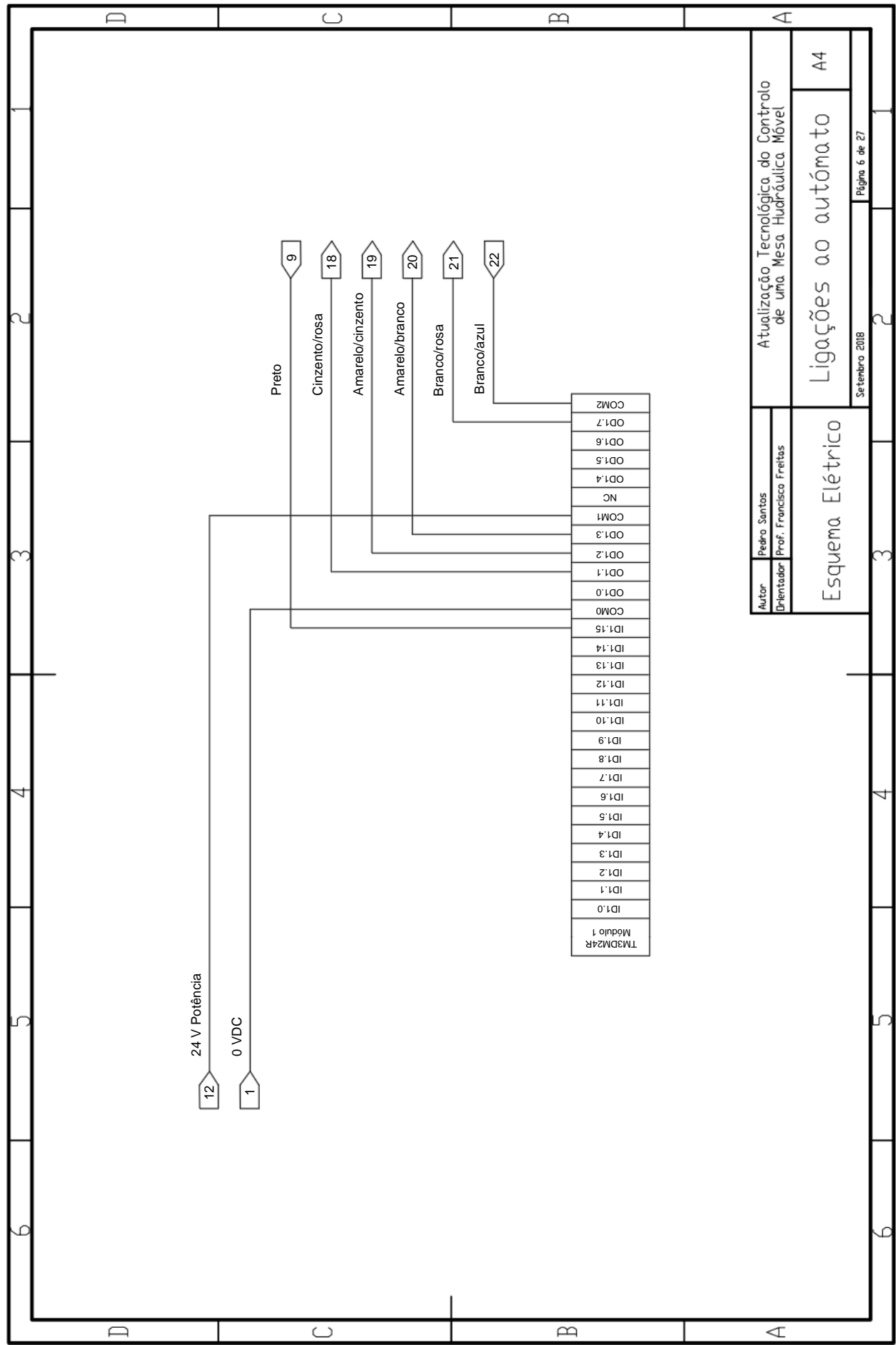


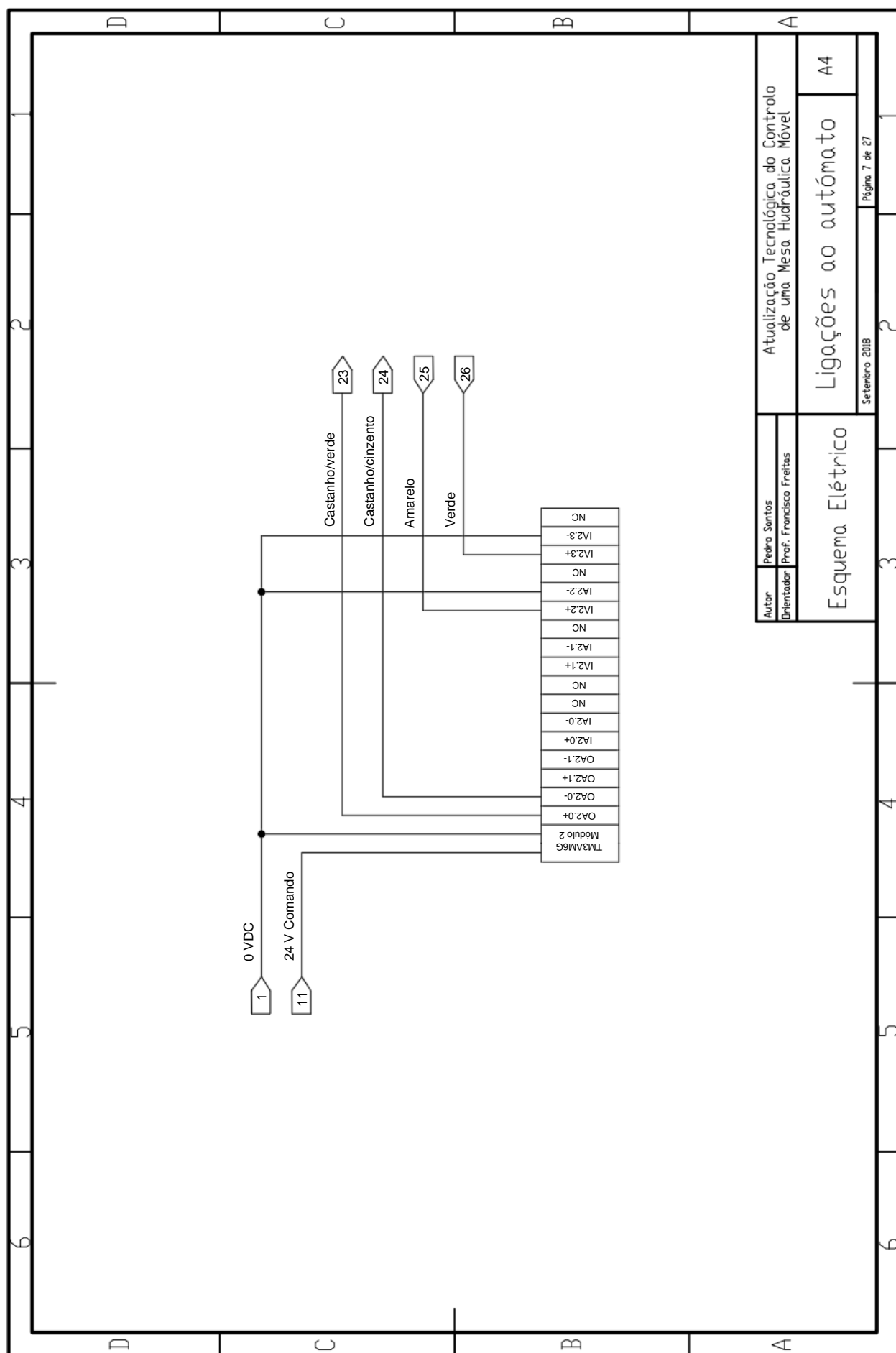
Autor	Pedro Santos	Atualização Tecnológica do Controle de uma Mesa Hidráulica Móvel	
	Orientador Prof. Francisco Freitas	Interruptores	
Esquema Elétrico		Setembro 2018	A4





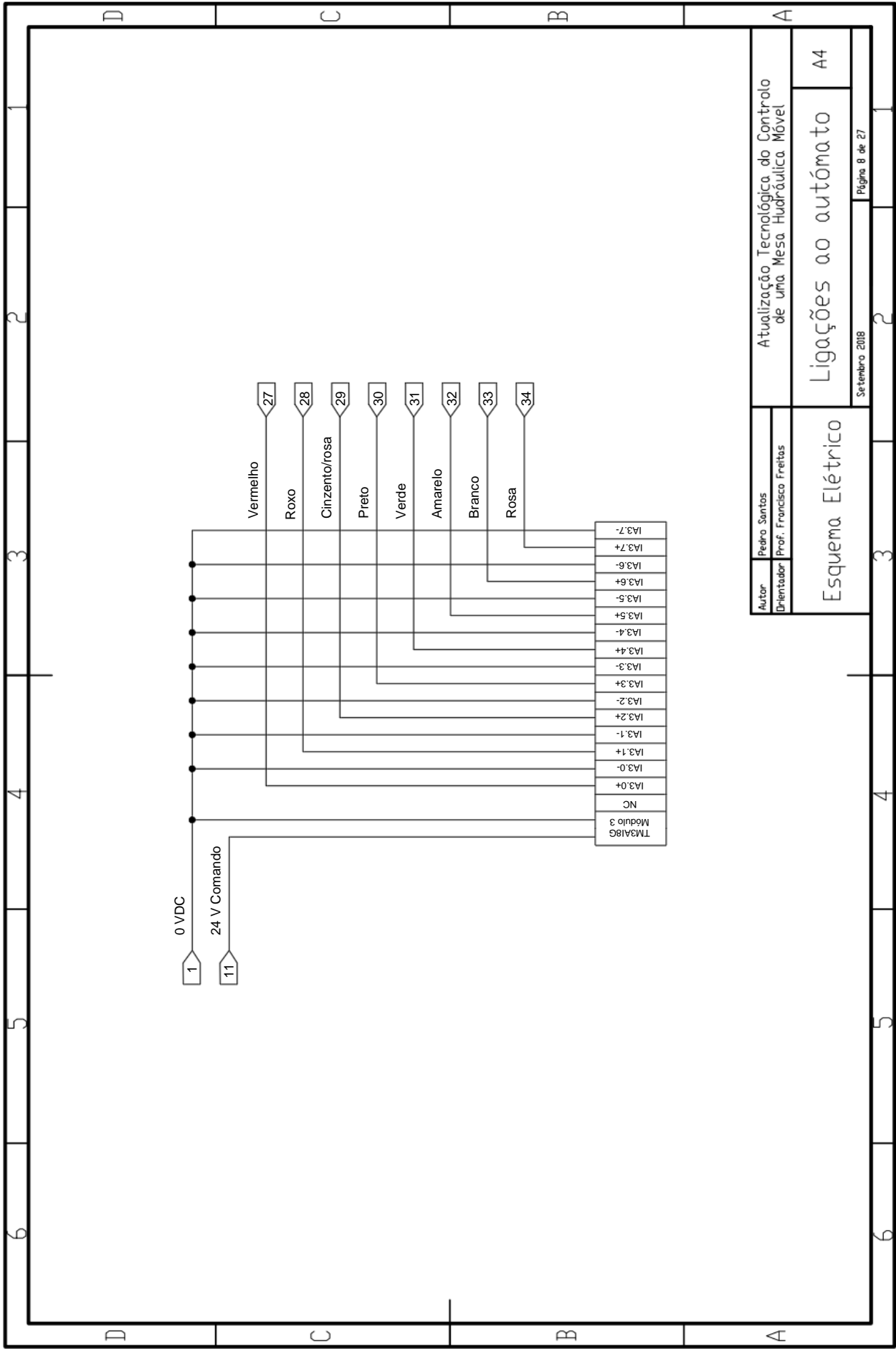
Autor	Pedro Santos	Atualização Tecnológica do Controle de uma Mesa Hidráulica Móvel	
	Prof. Francisco Freitas		
Esquema Elétrico		Ligações ao automático A4	
Setembro 2008		Página 5 de 27	





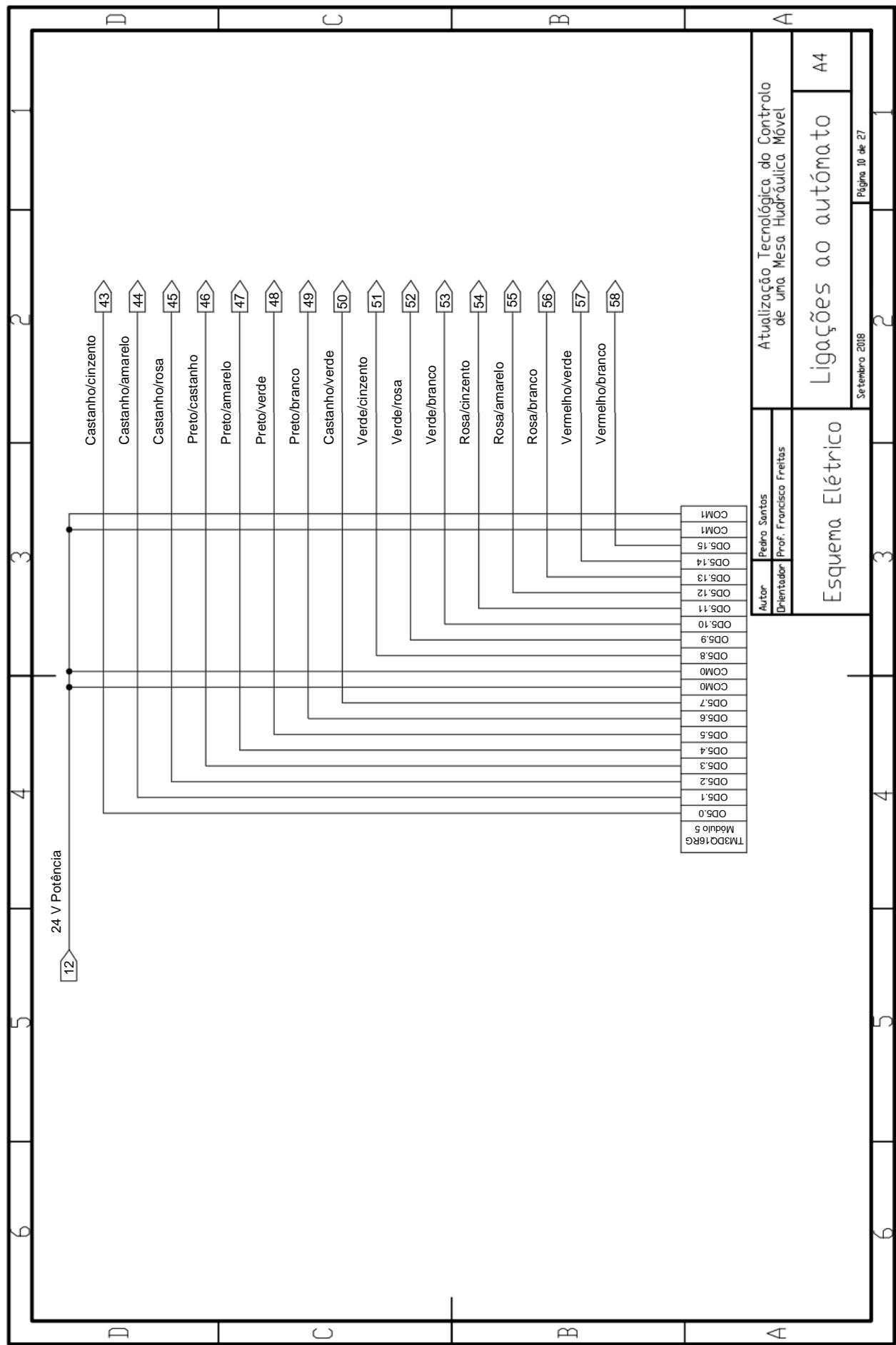
Autor	Pedro Santos	Atualização Tecnológica do Controle de uma Mesa Hidráulica Móvel	A4	
	Orientador			Prof. Francisco Freitas
Esquema Elétrico		Ligações ao automático		Setembro 2008

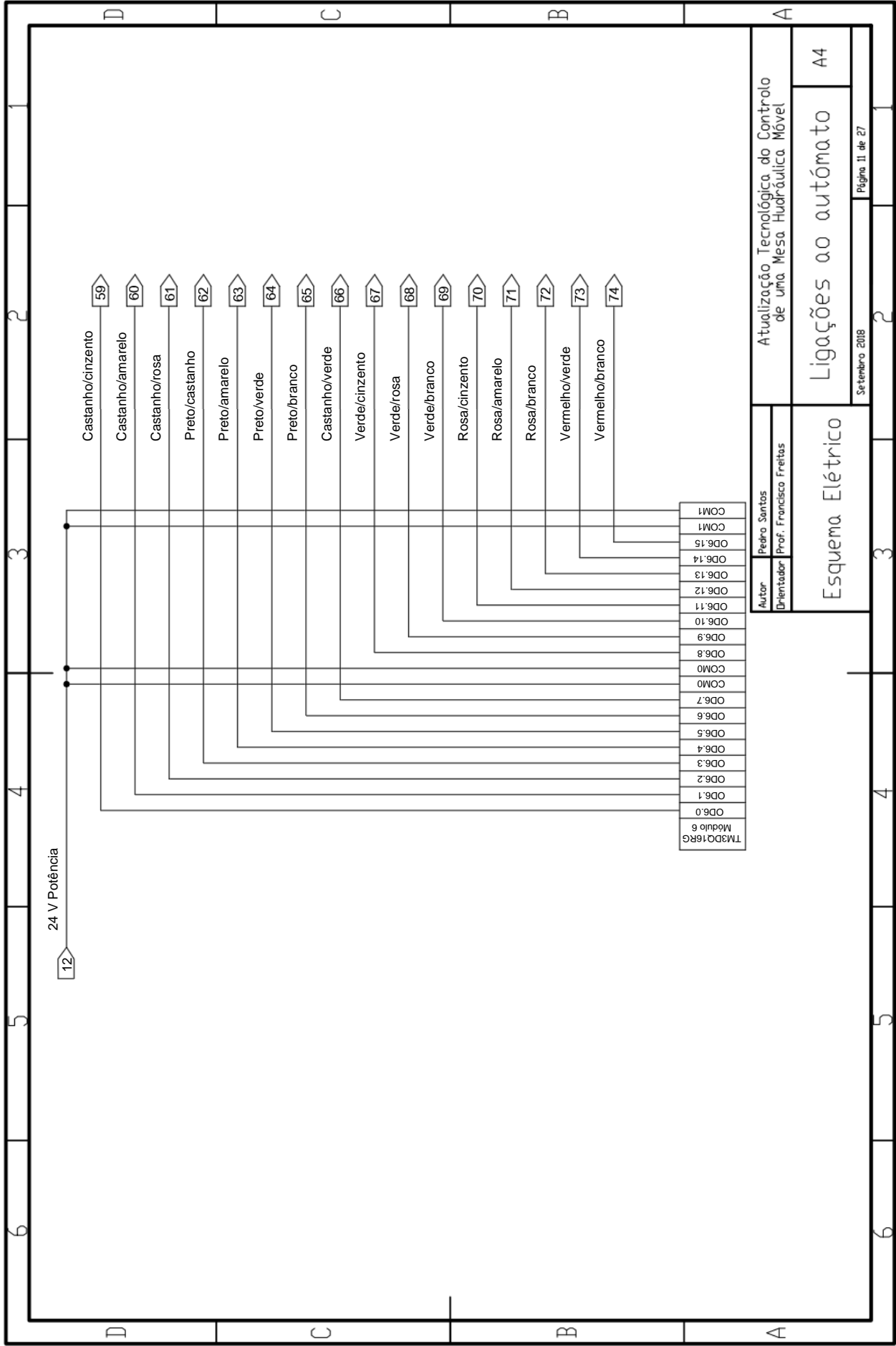
Página 7 de 27











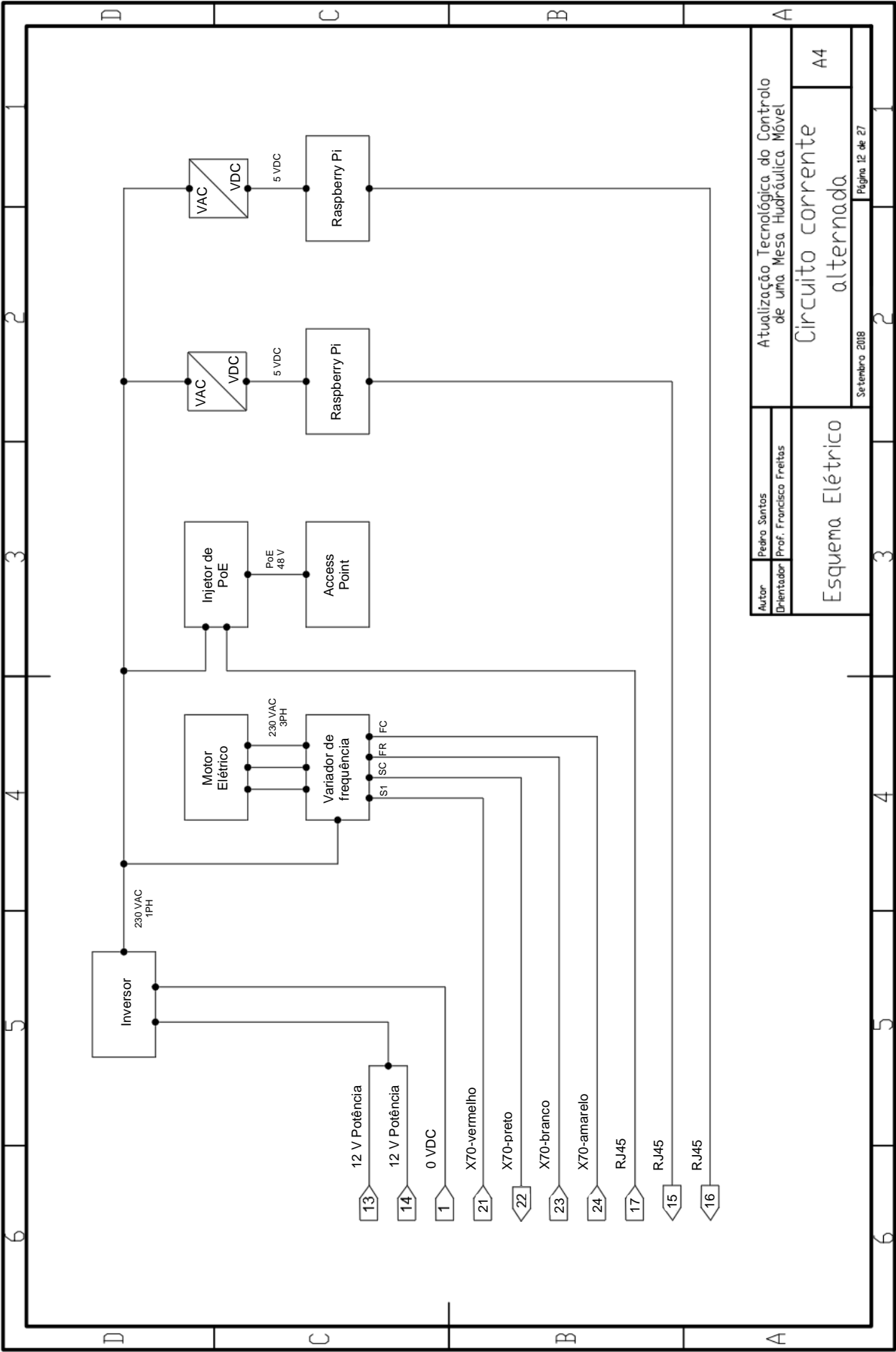
2

3

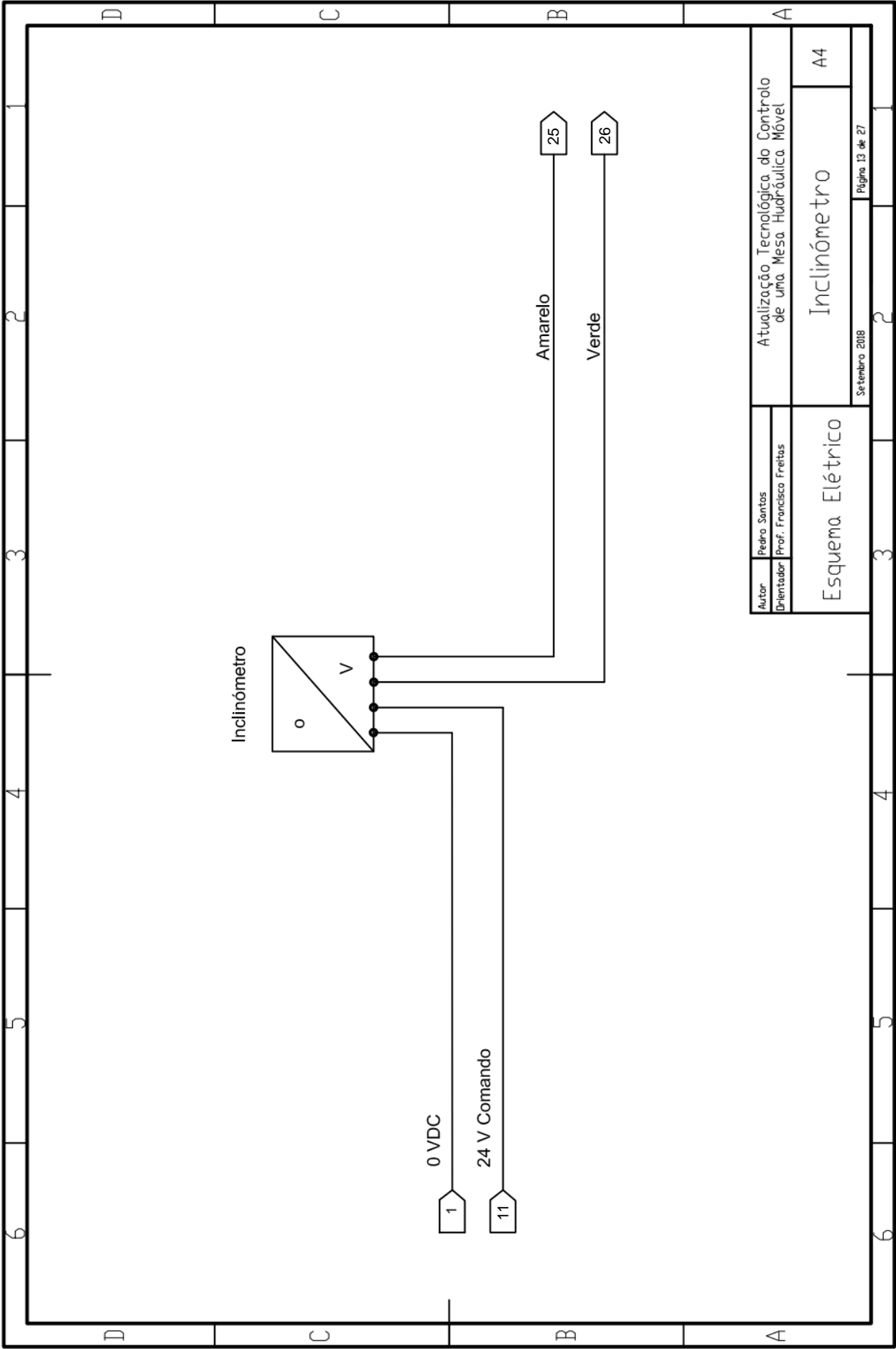
4

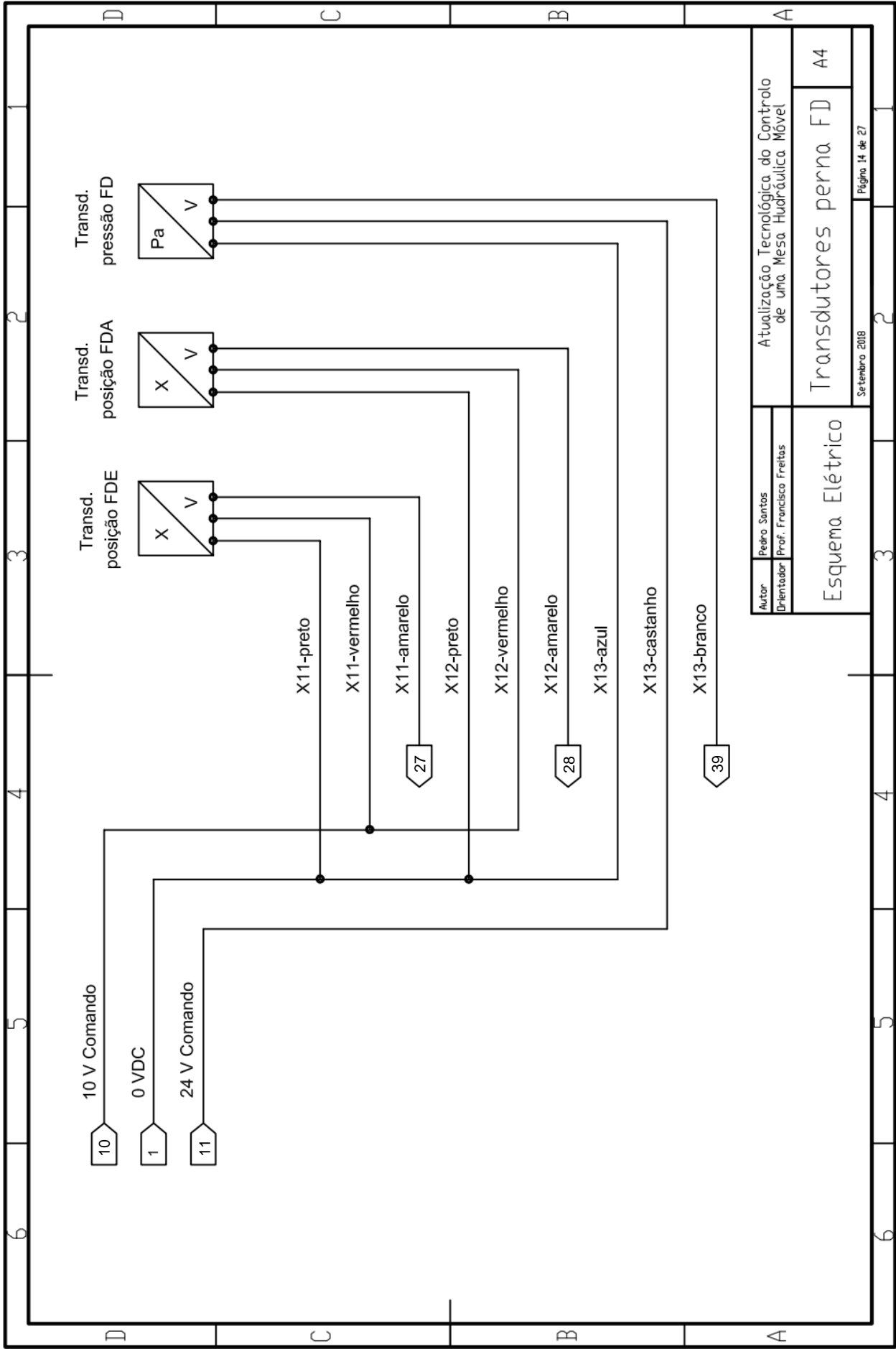
5

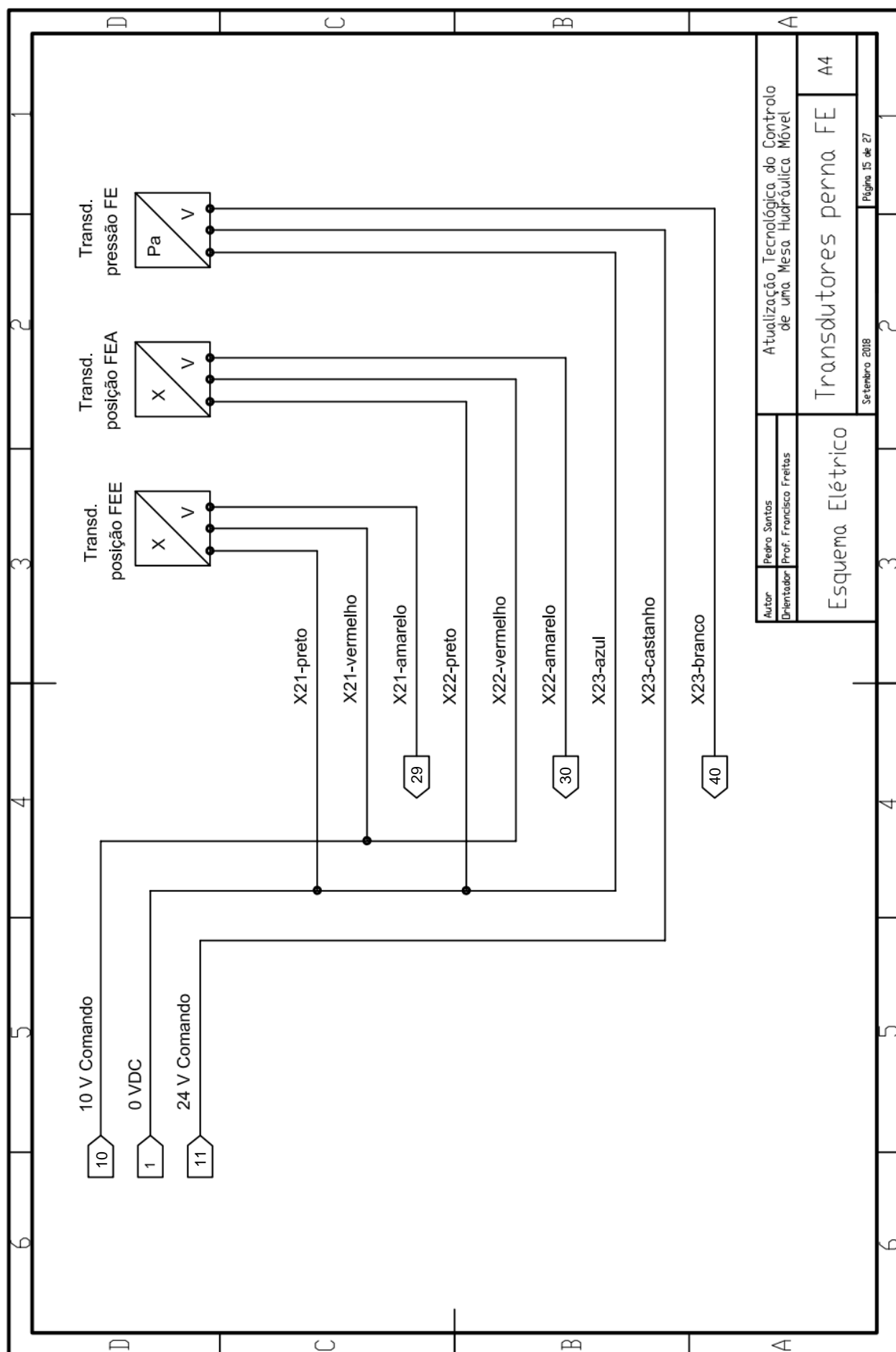
6



Autor Orientador	Pedro Santos Prof. Francisco Freitas	Atualização Tecnológica do Controle de uma Mesa Hidráulica Móvel	
		Circuito corrente alternada	A4
Esquema Elétrico		Setembro 2018	Página 12 de 27







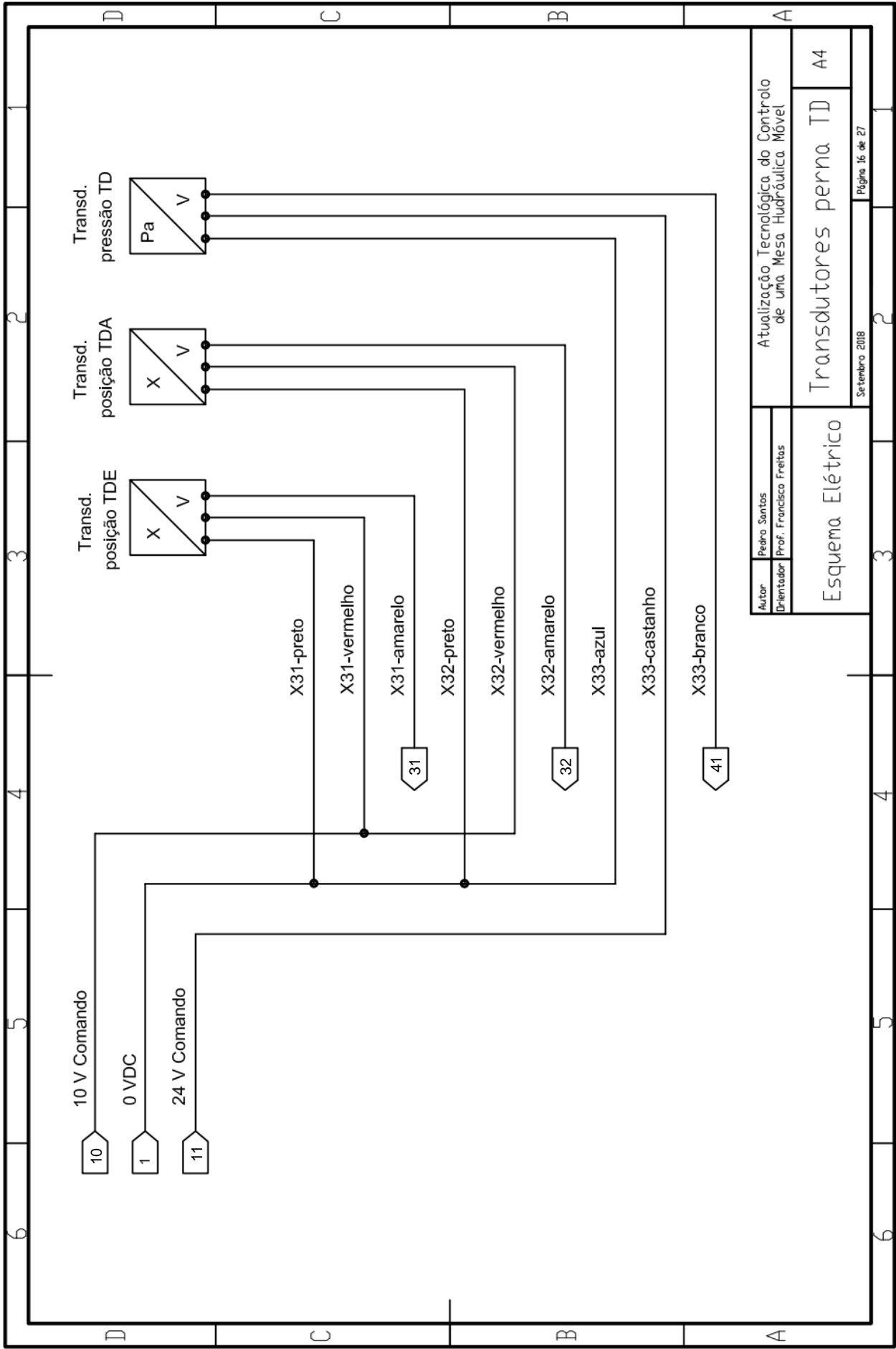
Atualização Tecnológica do Controle de uma Mesa Hidráulica Móvel

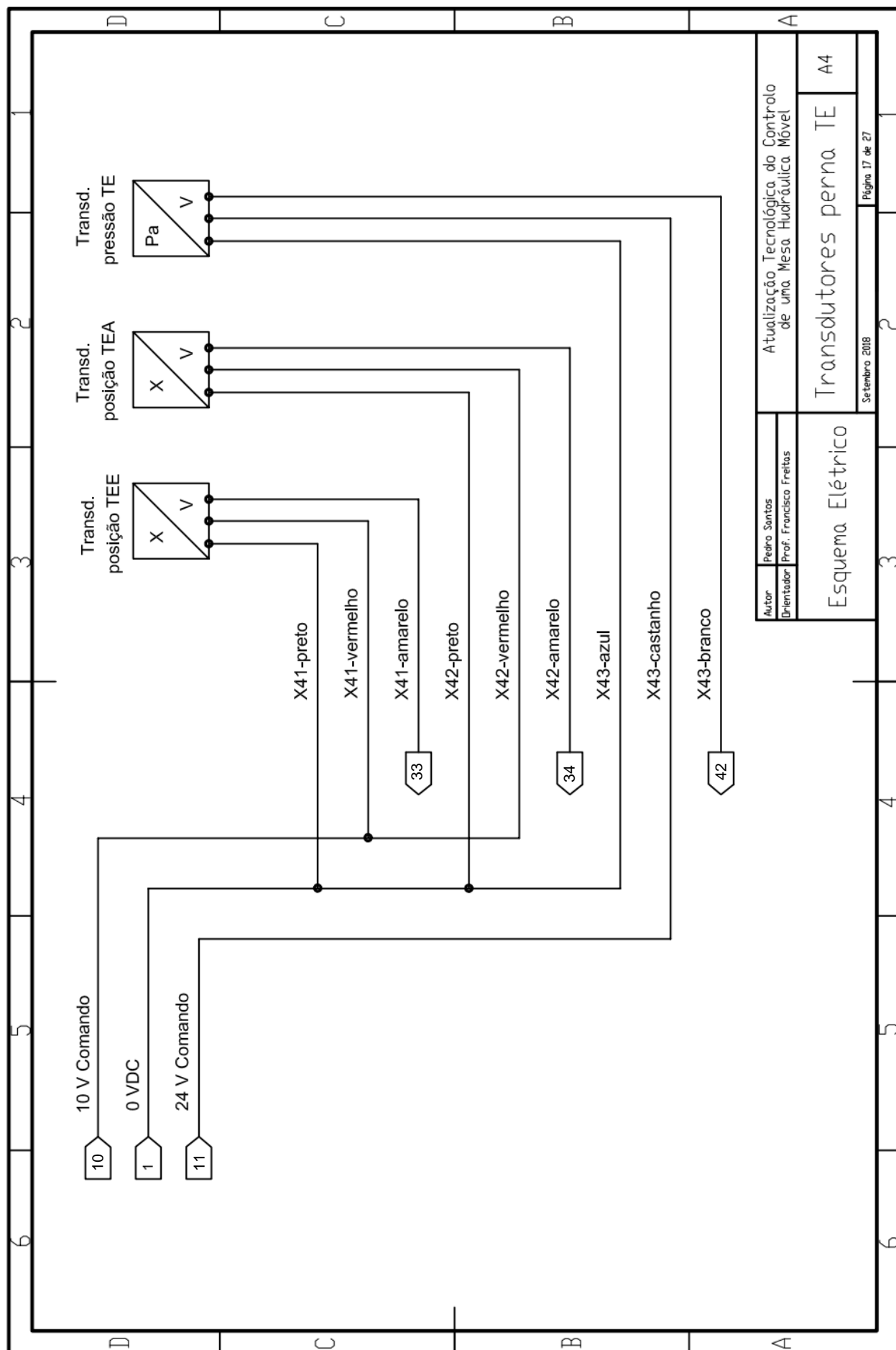
Esquema Elétrico

Transdutores perna FE A4

Sentembro 2008

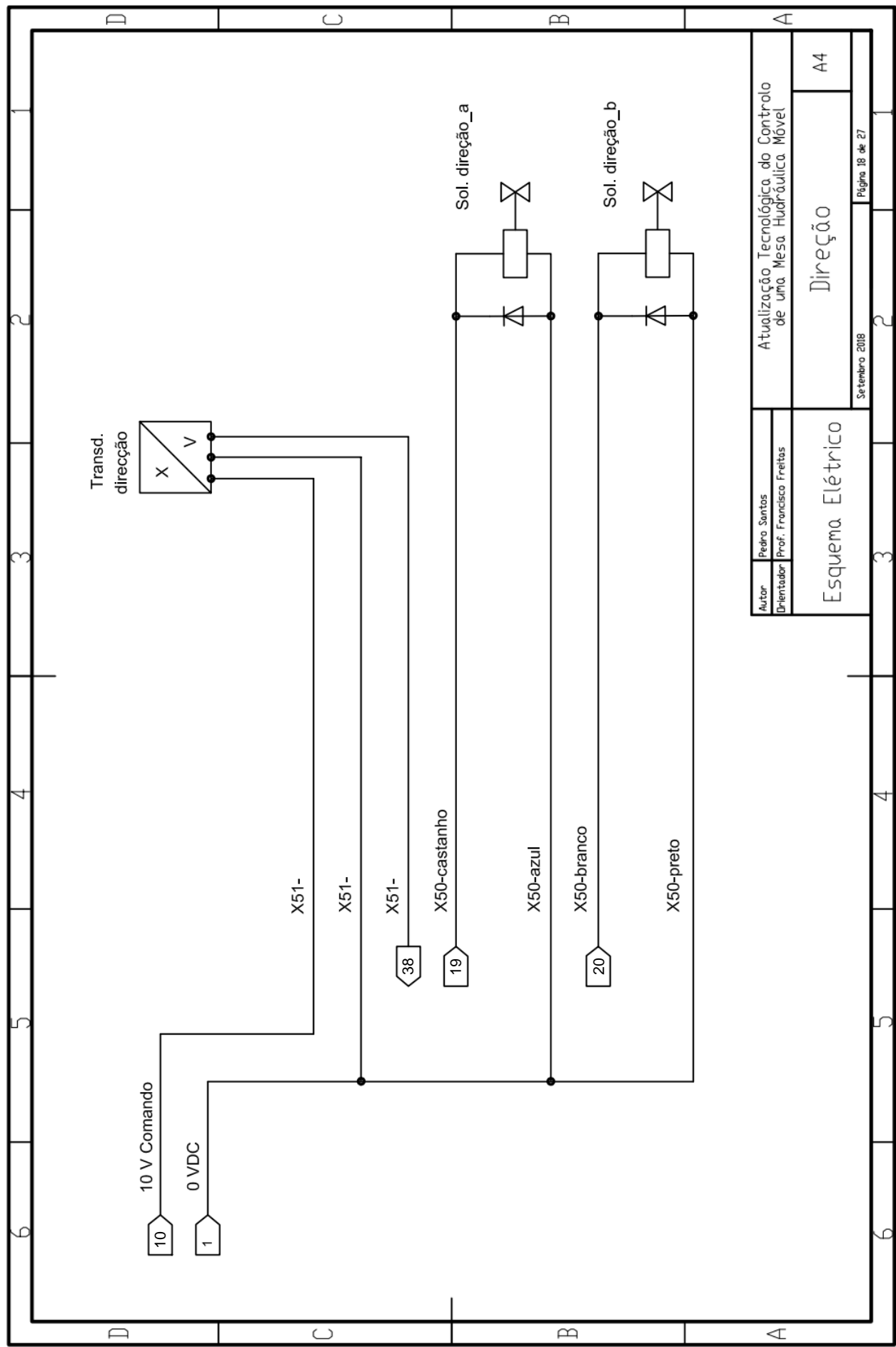
Página 15 de 27

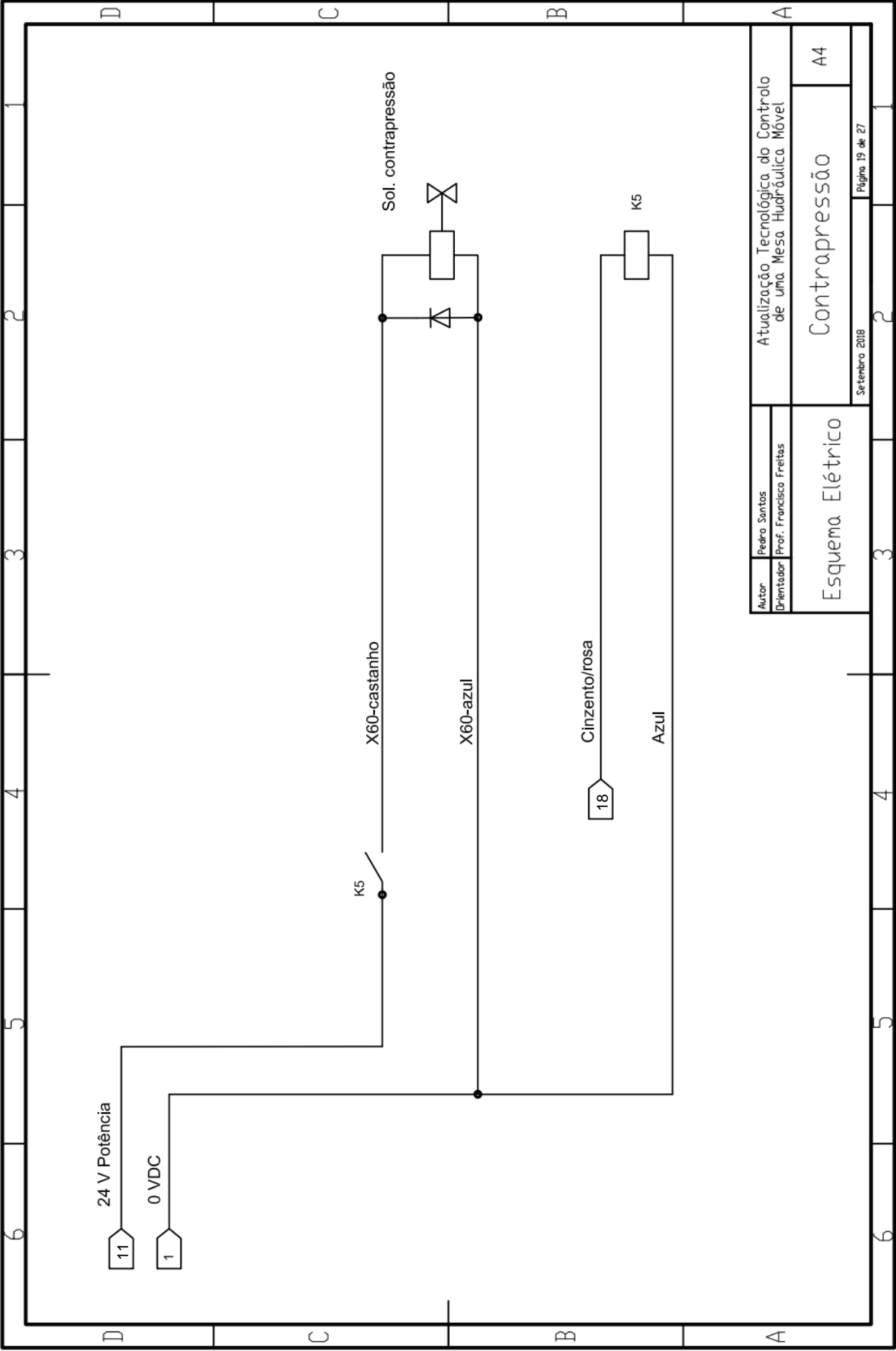


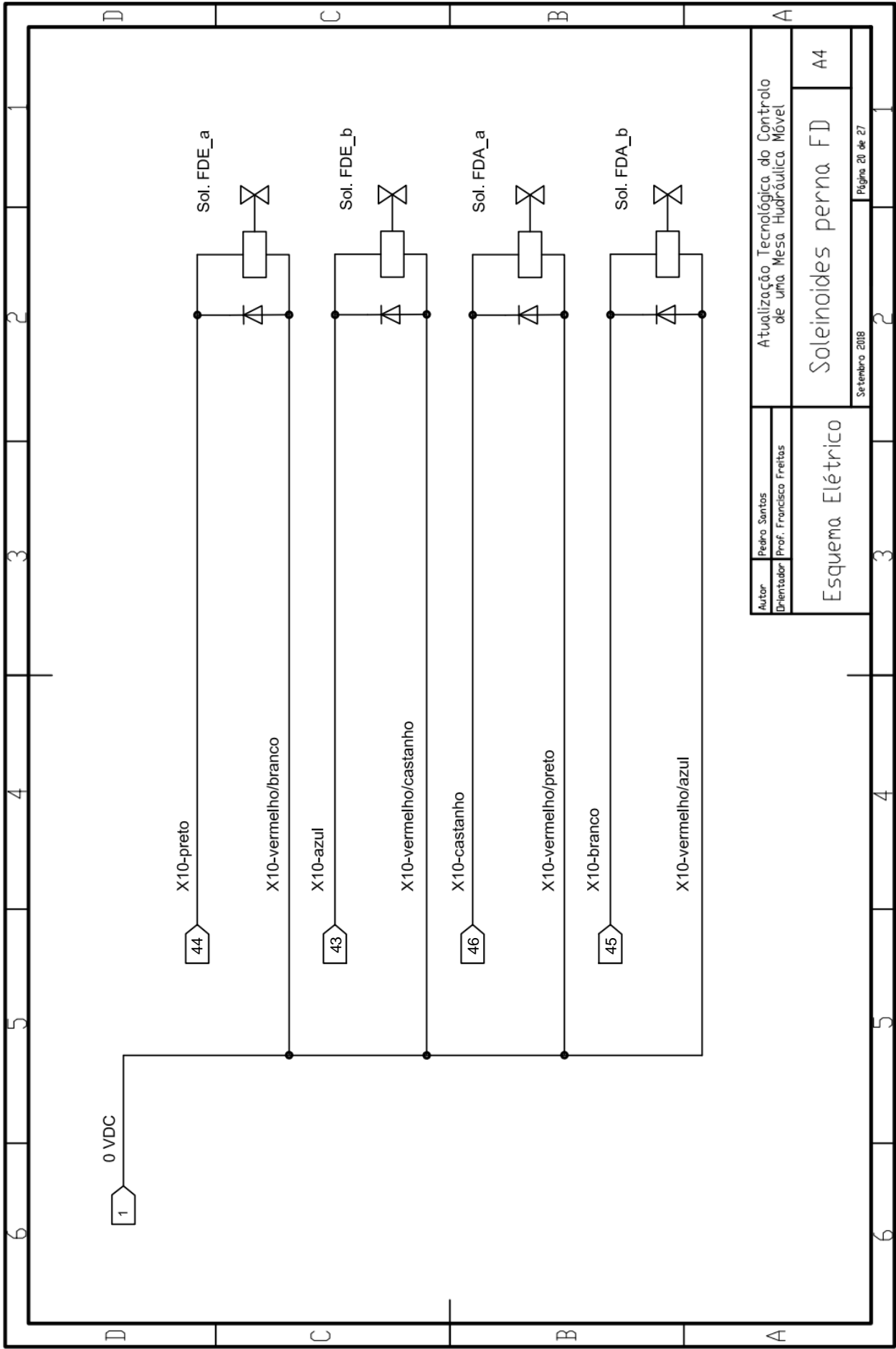


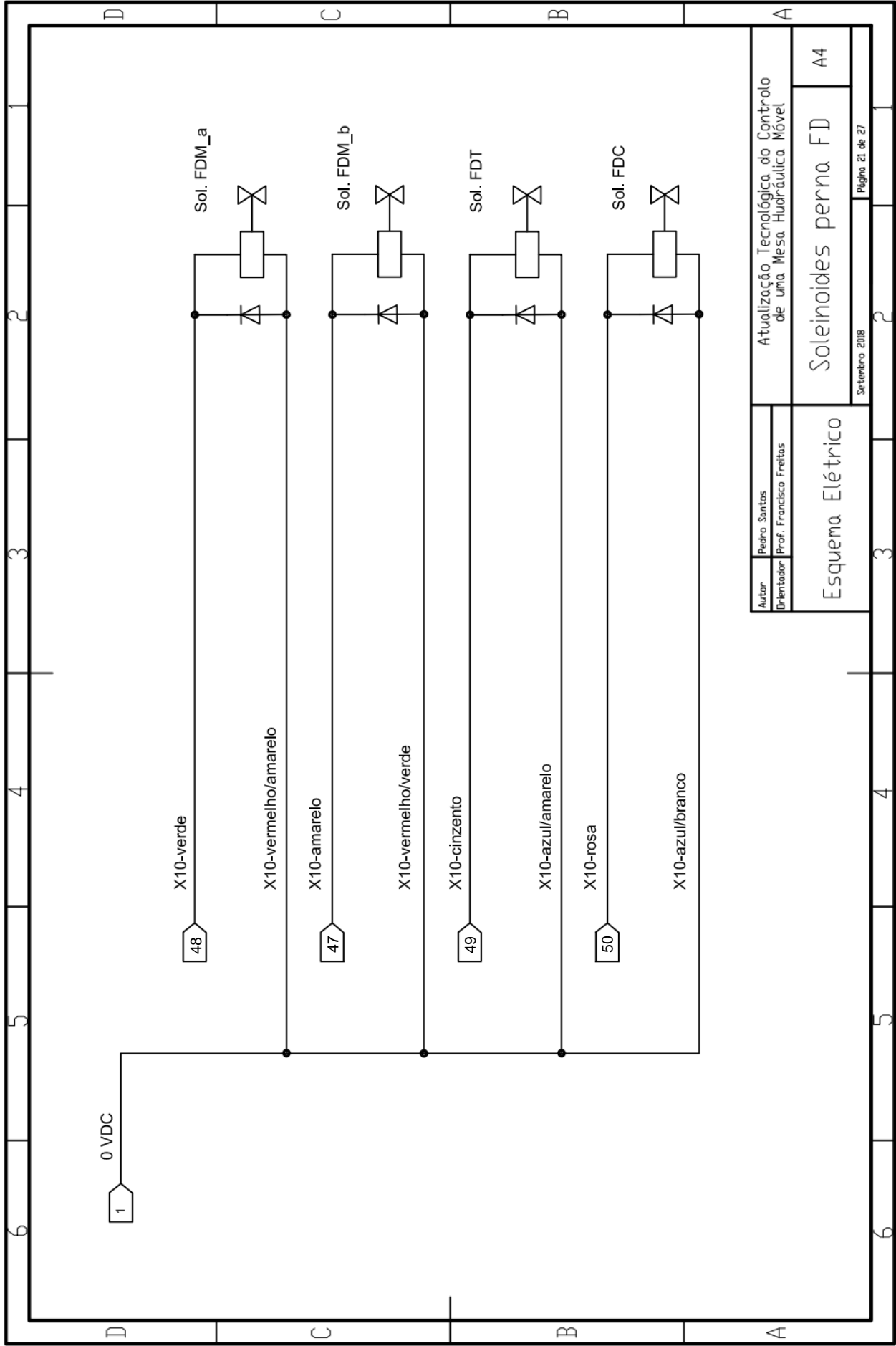
Autor Orientador	Pedro Santos	Atualização Tecnológica do Controle de uma Mesa Hidráulica Móvel	A4
	Prof. Francisco Freitas		
Esquema Elétrico		Transdutores perna TE	
		Setembro 2018	Página 17 de 27



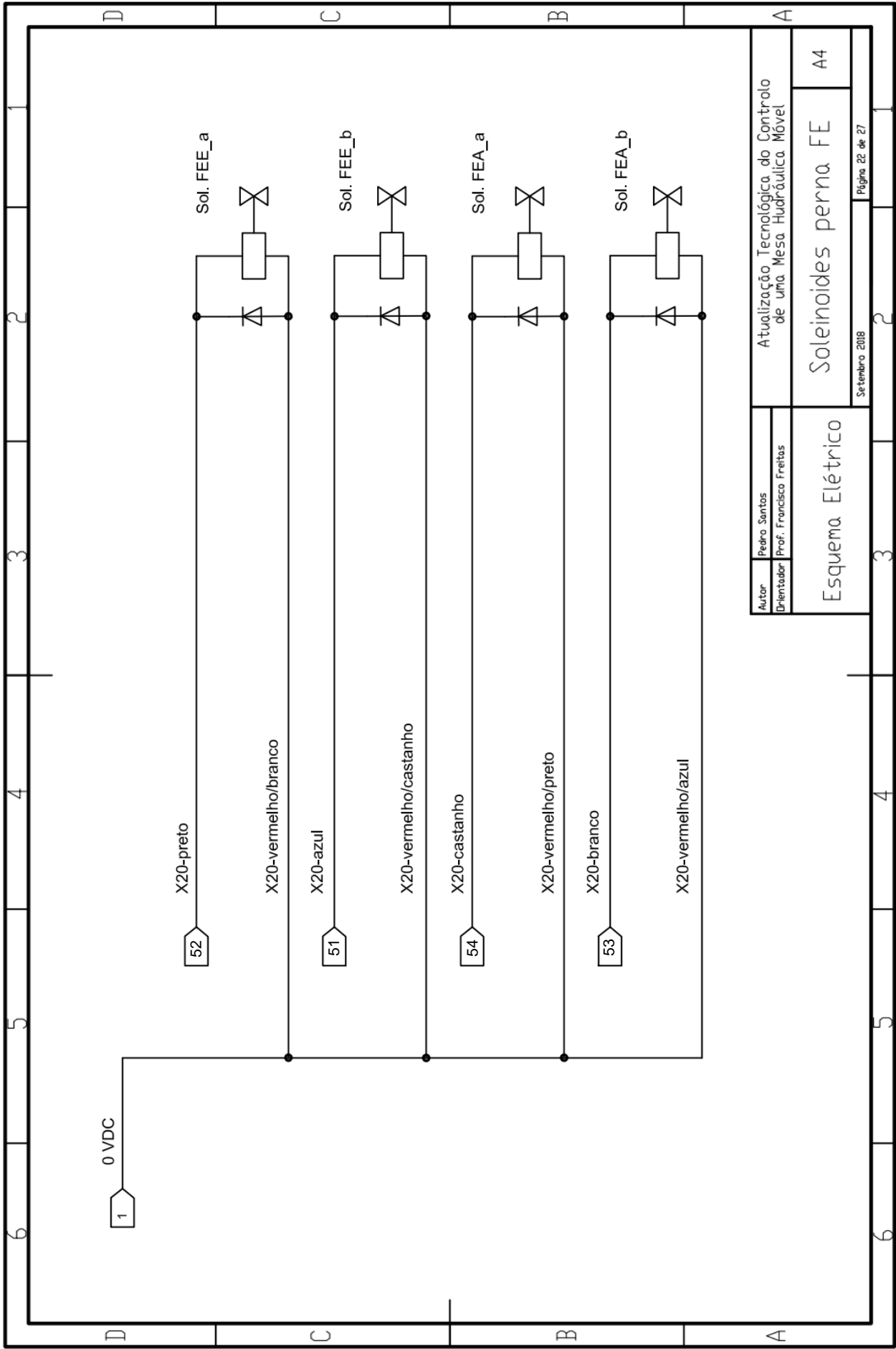


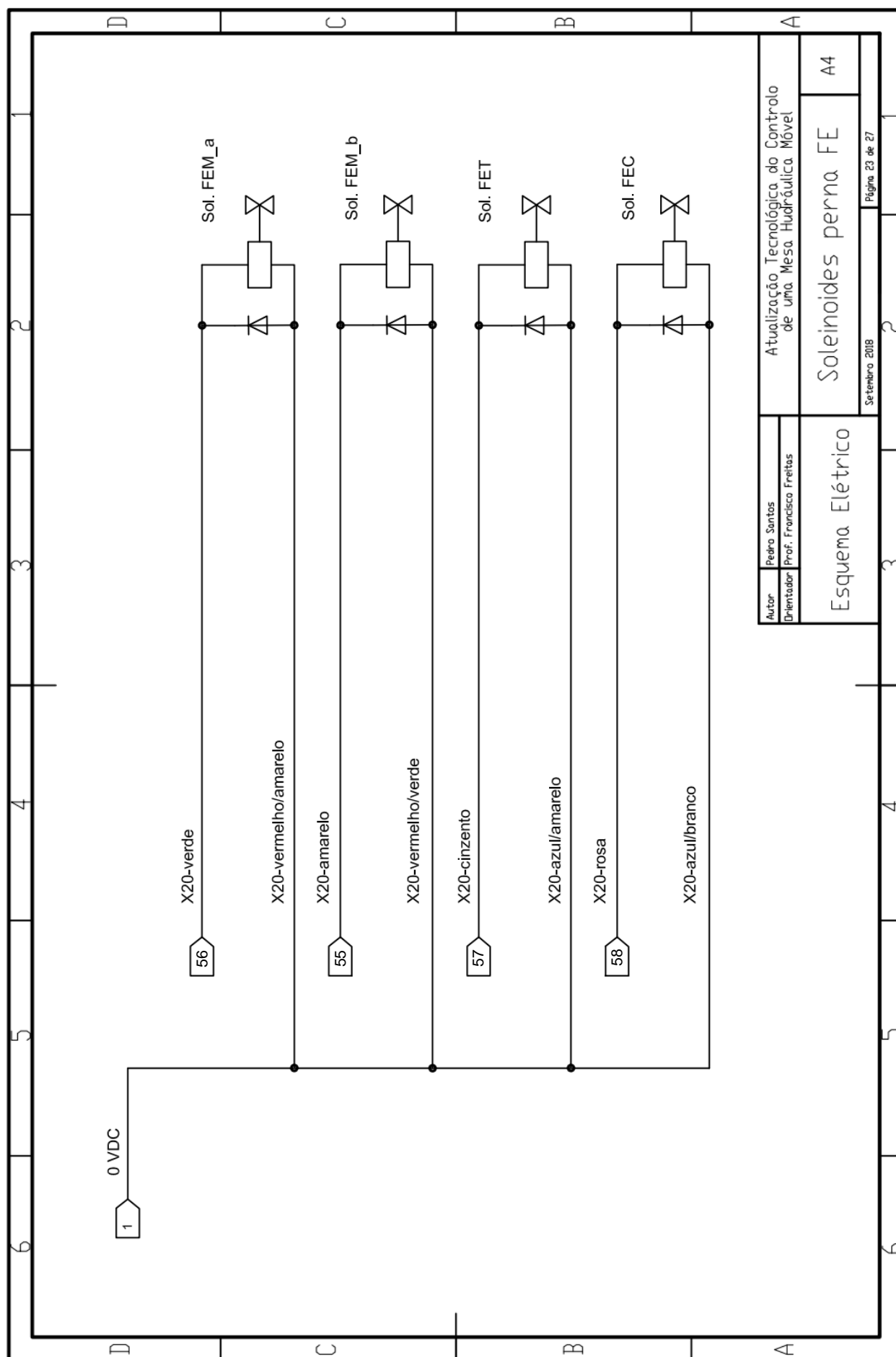




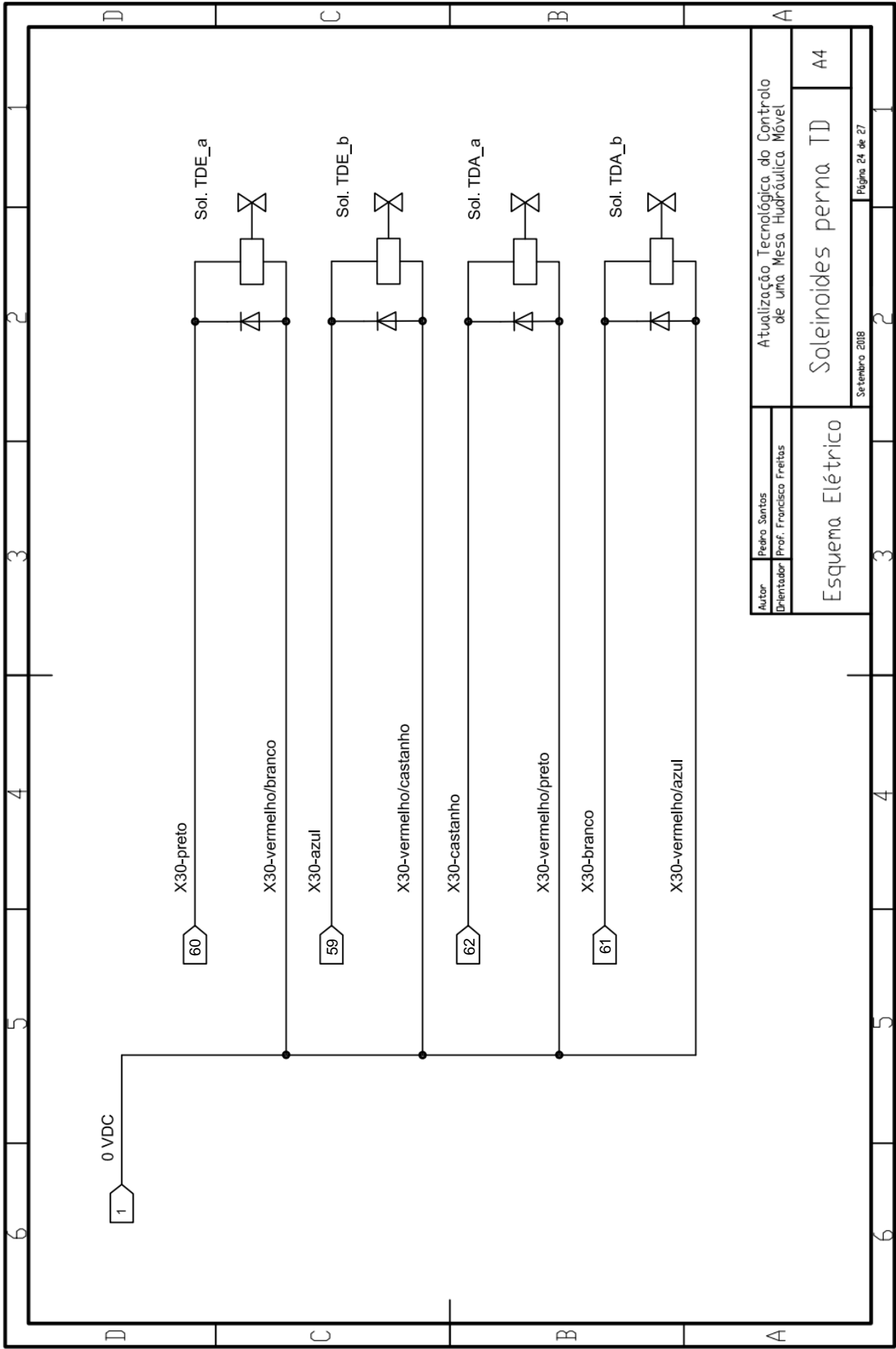


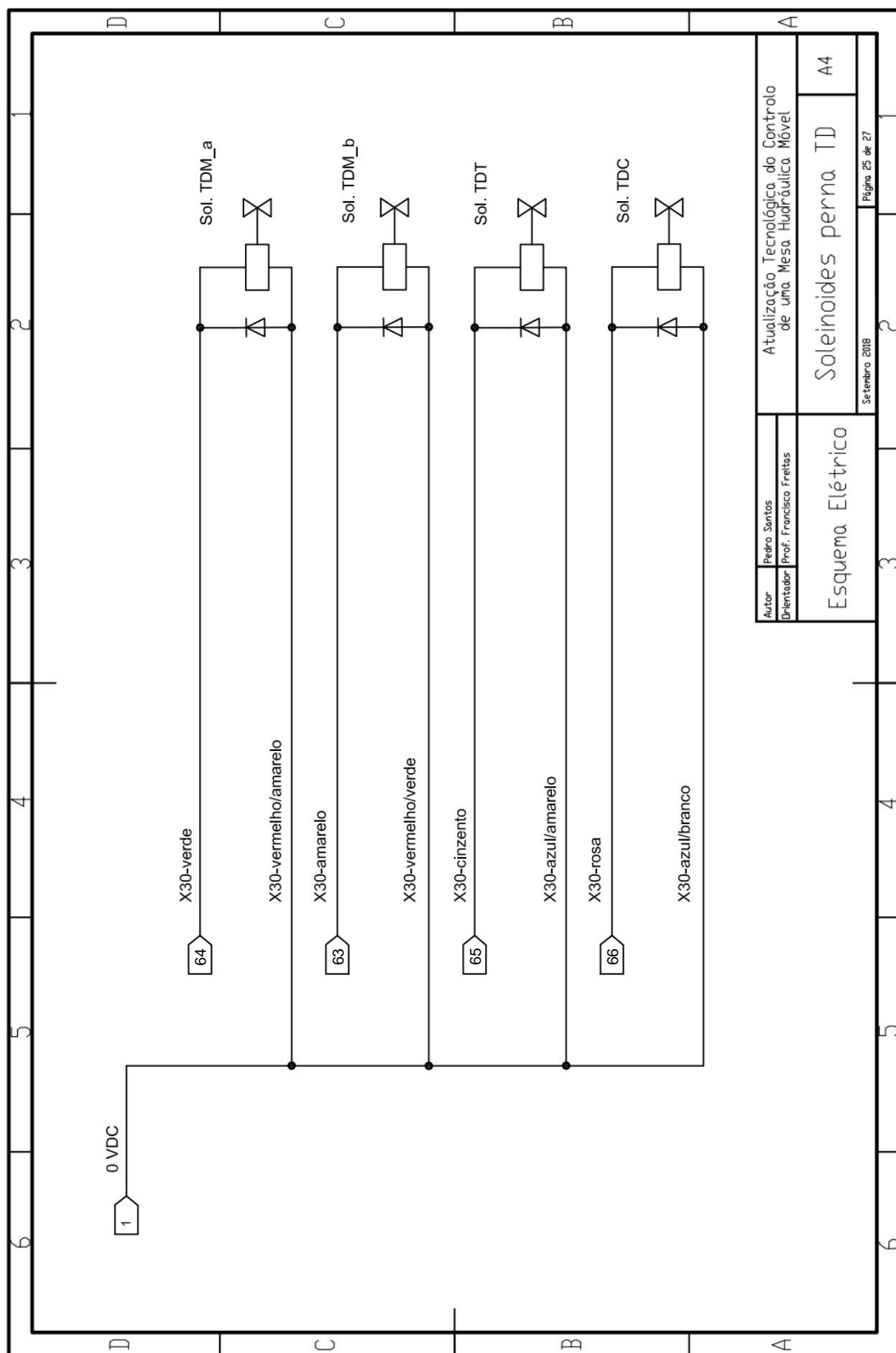
Autor Orientador	Pedro Santos	Atualização Tecnológica do Controle de uma Mesa Hidráulica Móvel	A4
	Prof. Francisco Freitas		
Esquema Elétrico		Soleinoides perna FD	
		Setembro 2008	Página 21 de 27





Autor	Pedro Santos	Atualização Tecnológica do Controle de uma Mesa Hidráulica Móvel	
	Orientador Prof. Francisco Freitas		
Esquema Elétrico		Solenoides perna FE A4	
		Setembro 2018	Página 23 de 27





Autor	Pedro Santos	Atualização Tecnológica do Controle de uma Mesa Hidráulica Móvel	
	Orientador Prof. Francisco Freitas		
Esquema Elétrico		Solenoides perna TD	
		A4	
		Setembro 2018	
		Página 25 de 27	



